Volume 2 · Number 1 · June 2026

Information Retrieval Research Journal
Published by RADBOUD UNIVERSITY PRESS
P.O. Box 9100, 6500 HA Nijmegen, The Netherlands
www.radbouduniversitypress.nl | radbouduniversitypress@ru.nl

ISSN: 3050-9106
E-ISSN: 3050-9114

Website: https://irrj.org

Cover design: Textcetera, The Hague
Print and distribution: Pumbo.nl

RADBOUD
UNIVERSITY
PRESS

Contents

Articles

Editorial

Mission & Scope

The Information Retrieval Research Journal (IRRJ) is a "diamond" open access journal that provides an international forum for the electronic and paper publication of high-quality scholarly articles in all areas of Information Retrieval. IRRJ commits to rigorous yet rapid reviewing. All published papers are freely available online. Final versions are published electronically immediately upon receipt with a DOI (ISSN 3050-9114). Paper volumes are published and sold by Radboud University Press (ISSN 3050-9106). IRRJ does not charge article processing costs and aims to support researchers from low-income countries that currently have a hard time engaging with the field. IRRJ seeks unpublished papers on information retrieval research grounded in statistics, machine learning, linguistics, the cognitive sciences, and perhaps other related research fields, such as recommender systems. Papers may contain:

- new principled algorithms with sound empirical validation, and with justification of mathematical, theoretical, or psychological nature;
- experimental and/or theoretical studies yielding new insight into the design and behavior of information retrieval systems, including user-centric studies;
- reproducibility studies and applications of existing techniques that highlight the strengths and weaknesses of current methods;
- formalization of new information retrieval tasks (e.g., in the context of new applications) and methods for assessing the performance of those tasks;
- new evaluation approaches, including responsible information retrieval, fairness and non-discrimination in search;
- review and survey papers that contribute to the understanding of the state of the art in information retrieval;
- resource papers that describe a test collection or labelled dataset, designs and protocols of evaluation tasks, or software tools and services.

Editorial board

Dmitry Ignatov (SE University, Russian Federation)
Udo Kruschwitz (University of Regensburg, Germany)
Saar Kuzi (Amazon, United States of America)
Matthew Lease (University of Texas at Austin, United States of America)
Siwei Liu (University of Aberdeen, United Kingdom)
Graham McDonald (University of Glasgow, United Kingdom)
Thomas Mandl (University of Hildesheim, Germany)
Bruno Martins (University of Lisbon, Portugal)
Parth Mehta (Parmonic, United States of America)
Franco Maria Nardini (ISTI-CNR, Pisa, Italy)
Elham Naghizade (RMIT University, Australia)
Sachin Pathiyan Cherumanal (RMIT University, Australia)
Javier Parapar (University of Coruña, Spain)
Benjamin Piwowarski (CNRS, Sorbonne Université, France)
Alisa Rieger (GESIS – Leibniz Institute for the Social Sciences, Germany)
Jialie Shen (City St George's University of London, United Kingdom)
Farhad Shokraneh (University of Oxford, United Kingdom)
Damiano Spina (RMIT University, Australia)
Mark Stevenson (University of Sheffield, United Kingdom)
Nicola Tonellotto (University of Pisa, Italy)
Md Zia Ullah (Edinburgh Napier University, United Kingdom)
Sanne Vrijenhoek (University of Amsterdam, The Netherlands)
Yumeng Wang (Leiden University, The Netherlands)
Anna Wróblewska (Warsaw University of Technology, Poland)
Siqi Yi (University of Texas at Austin, United States of America)

Advisory board

Paul Kantor (Emeritus, Rutgers University, United States of America)
Stephen Robertson (formerly Microsoft Research, United Kingdom)

Production editor

Hanan Noij (Radboud University Press, The Netherlands)

Webmaster

Kay Pepping (KNAW, The Netherlands)

IRRJ publications in previous issues

Paul Kantor. Editorial: AI Resilience and Information Retrieval. *IRRJ* 1(2), 339–343, 2025. doi: 10.54195/irrj.25004.

Bhaskar Mitra. Emancipatory Information Retrieval. *IRRJ* 1(2), 313–338, 2025. doi: 10.54195/irrj.24531.

Meng Yuan, Lida Rashidi, Justin Zobel. Exploring Embedding Interpretability by Correspondences Between Topic Models and Text Embeddings. *IRRJ* 1(2), 281–312, 2025. doi: 10.54195/irrj.23703.

Massimo Melucci. An Eigensystem for Topic Term Weighting yields Fair and Effective Document Rankings. *IRRJ* 1(2), 247–280, 2025. doi: 10.54195/irrj.23480.

Madhukar Dwivedi, Jaap Kamps. Effectiveness of In-Context Learning for Due Diligence: A Reproducibility Study of Identifying Passages for Due Diligence. *IRRJ* 1(2), 221–245, 2025. doi: 10.54195/irrj.22626.

Catarina Pires, Sérgio Nunes, Luís F. Teixeira. Evaluating Dense Model-based Approaches for Multimodal Medical Case Retrieval. *IRRJ* 1(2), 197–220, 2025. doi: 10.54195/irrj.19769.

Yue Zheng, Haiming Liu, Mike Wald. A survey of Inclusive Information Access. *IRRJ* 1(2), 167–196, 2025. doi: 10.54195/irrj.21092.

Emma J. Gerritse, Faegheh Hasibi , Arjen P. de Vries. Graph Embeddings to Empower Entity Retrieval. *IRRJ* 1(1), 137–165, 2025. doi: 10.54195/irrj.19877.

Charles Clarke. Annotative Indexing. *IRRJ* 1(1), 109–136, 2025. doi: 10.54195/irrj.19910.

Sameh Frihat, Norbert Fuhr. Supporting Evidence-Based Medicine by Finding Both Relevant and Significant Works. *IRRJ* 1(1), 93–108, 2025. doi: 10.54195/irrj.19784.

Bhaskar Mitra. Search and Society: Reimagining Information Access for Radical Futures. *IRRJ* 1(1), 47–92, 2025. doi: 10.54195/irrj.19654.

Ian Soboroff. Don't Use LLMs to Make Relevance Judgments. *IRRJ* 1(1), 29–46, 2025. doi: 10.54195/irrj.19625.

Savvina Daniil, Manel Slokom, Mirjam Cuper, Cynthia Liem, Jacco van Ossenbruggen, Laura Hollink. On the challenges of studying bias in Recommender Systems: The effect of data characteristics and algorithm configuration. *IRRJ* 1(1), 3–27, 2025. doi: 10.54195/irrj.19607.

Djoerd Hiemstra, Ismail Sengor Altingovde, Solomon Atnafu, Daniela Godoy, Ben He, Makoto Kato, Shangsong Liang, Haiming Liu, Vanessa Murdock, Monica Paramita, Barbara Poblete, Negin Rahimi, Debarshi Kumar Sanyal, Johanne Trippas. Editorial. *IRRJ* 1(1), 1–2, 2025. doi: 10.54195/irrj.23259.

Information Retrieval Research 2 (2026) 1-33 Submitted 7/25; Revised: 1/26; Published 2/26

CRAWLDoc: A System for Contextual Ranking and Bibliographic Metadata Extraction from Web Resources

Fabian Karl FABIAN.KARL@UNI-ULM.DE
Universität Ulm, Germany

Ansgar Scherp ANSGAR.SCHERP@UNI-ULM.DE
Universität Ulm, Germany

Editor: Makoto Kato

Abstract

Publication databases rely on accurate metadata extraction from diverse web sources, yet variations in web layouts and data formats present challenges for metadata providers. This paper introduces CRAWLDoc, a novel system for contextual ranking of linked documents and metadata extraction from web resources. Using a publication's DOI, CRAWLDoc retrieves the landing page and associated linked documents, including PDFs, ORCID profiles, and supplementary materials. It embeds these documents, along with anchor texts and URLs, into a unified representation. Our layout-independent embedding and ranking system ensures robustness across various web layouts and formats. Experimental results show that CRAWLDoc improves bibliographic metadata extraction compared to relying solely on landing pages. In document ranking, our fine-tuned dense retriever outperforms sparse baselines such as BM25 and BM25+. A leave-one-out experiment across six publishers indicates robustness across publisher-specific layouts. Our source code and dataset can be accessed at `https://github.com/FKarl/crawldoc-metadata-extraction`

Keywords: Large Language Models, Document Ranking, Bibliographic Metadata Extraction, Scholarly Dataset

1 Introduction

Databases such as Web of Science[1], Crossref[2], and DBLP[3] are crucial academic resources of bibliographic information. Extracting high-quality metadata about new publications, such as authors and affiliations, is essential for these services. While there are methods and tools for extracting bibliographic metadata (Lopez, 2009; Tkaczyk et al., 2015), these are typically restricted to a single document like a PDF. Beyond curated bibliographies, scholarly search engines and metadata aggregators (e. g., Google Scholar, Semantic Scholar, OpenAlex) must integrate evidence from heterogeneous web sources and formats, making robustness to layout and representation differences a practical requirement. Currently, many potential sources of web content that may contain valuable metadata are underutilized. These include full texts, publication PDFs, conference websites, publisher landing pages, ORCIDs, and other web content. One reason is the high heterogeneity of these sources, which complicates metadata extraction and integration.

1. `https://www.webofscience.com`
2. `https://www.crossref.org`
3. `https://dblp.org`

 WWW: https://irrj.org DOI: 10.54195/irrj.23861

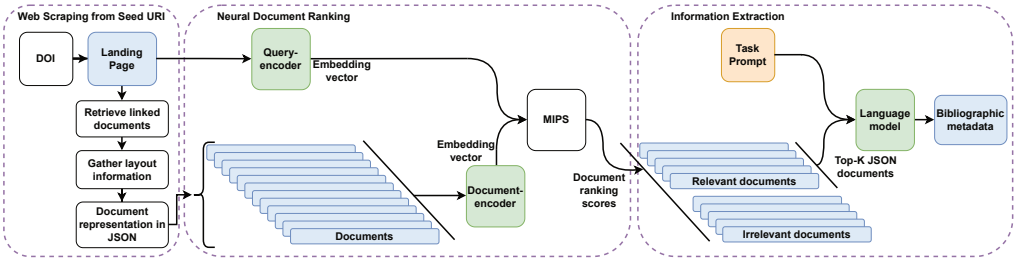


Figure 1: Illustration of the task of bibliographic information extraction from heterogeneous web sources. The process starts with an input DOI, which is resolved to access the associated landing page. We then employ a document-as-query approach to rank the documents linked from the landing page. A maximum inner product search (MIPS) ranks the top-k documents based on embeddings from a small language model. Finally, from the selection of k relevant documents, a generative large language model extracts the desired bibliographic metadata.

1.1 The DBLP Computer Science Bibliography Scenario

We consider the example of DBLP, the de facto main metadata provider in computer science. The main strategy for integrating publisher-provided metadata is to implement publisher-specific wrappers, an approach that is time-consuming and requires maintenance whenever the publisher changes its website (Schenkel, 2018). Thus, there is a need for an automated service to systematically search for and extract bibliographic metadata from multiple web sources. Often, the bibliographic information cannot be found on a single website, e. g., the publication's landing page, necessitating to harvest linked documents and finding those relevant to the publication. Identifying relevant linked documents is challenging since two web documents with similar layouts and text can refer to different papers, with paper-specific components like titles, authors, and affiliations. Another challenge is the heterogeneity of web data. Important documents can be in HTML or other formats like PDF.

Among bibliographic metadata, affiliations are a key challenge. They are frequently missing from landing pages and metadata exports, and when present they often require correctly assigning multiple affiliation strings to individual authors. Therefore, a key challenge is to improve the extraction of affiliation information while maintaining high precision for existing bibliographic metadata, such as titles and author names. Affiliation metadata is particularly important for bibliometric analyses, funding attribution, and institutional reporting, yet it is the most difficult to extract accurately. Although one might assume that publisher-provided bibliographies, for instance through "Export BibTeX" on DBLP or via APIs like Zotero or arXiv, would suffice, affiliation details are not consistently included. Some sources occasionally provide affiliation data, but often not on a per-paper basis.

A method is needed to reliably extract affiliation information on a per-paper level and for each author individually. For instance, imagine a paper with five authors, three sharing one institution and a fourth from another, while the fifth is affiliated with both. The correct assignment of each affiliation to the right author remains unresolved, partly due to the various layouts in PDFs and websites. Paper templates can locate affiliations in footnotes or margins, referencing them with symbols or superscripts that complicate automated extrac-

tion. Another reason is that using wrappers or APIs relies on crawling publisher websites, which is expensive to maintain (Schenkel, 2018).

1.2 Proposed Solution CRAWLDoc

We propose a novel retrieval system CRAWLDoc (Contextual RAnking of Web-Linked Documents), see Figure 1, that can automatically identify relevant data sources and extract bibliographic metadata from diverse web sources. Input is a Digital Object Identifier (DOI)[4] of a publication, which is provided by publishers (Ley, 2009). The web content linked from this seed URI is harvested and analyzed. We employ a document-as-query approach to identify relevant linked content that refers to the same paper as the DOI and may carry relevant metadata. To this end, we embed the source document and linked documents along with their associated anchor texts and URLs into a shared vector space and treat the publication's landing page as the query. A ranking is computed by the similarity between the landing page embedding and the embeddings of linked documents, effectively identifying the most relevant sources for metadata extraction. By embedding the content, we effectively address the challenge that the web sources from which the data is extracted are highly diverse (Schenkel, 2018) and vary in structure and format.

The key difference between CRAWLDoc and existing approaches such as Enlil (Do et al., 2013), CERMINE (Tkaczyk et al., 2015), or GROBID (Lopez, 2009) lies in their input scope. While these systems process a single PDF document, CRAWLDoc operates on multiple documents of heterogeneous formats. Specifically, CRAWLDoc first collects documents from the 1-hop neighborhood of the input DOI/URI. Subsequently, we rank these documents using neural embeddings to identify the most relevant sources, and then apply an off-the-shelf information extraction component on the top-ranked documents. In our case, we use GPT-4o as the extractor.

We evaluate CRAWLDoc on a newly derived dataset from DBLP, comprising 600 publications from the six largest computer science publishers. Our dataset is unique as it provides manually annotated relevancy labels for all outgoing links from publication landing pages along with bibliographic metadata including titles, years, authors' names, and affiliations. Our experiments show that CRAWLDoc reliably identifies relevant web documents based on a single seed document. A zero-shot language model proficiently extracts bibliographic metadata when given the correct context. CRAWLDoc consistently improves the extraction performance for all publishers compared to a naive approach of solely using the landing page. A leave-one-out experiment shows that our system is robust w. r. t. the extraction from websites with various layouts from publishers that were not part of the training dataset. In summary, our contributions are:

- A document-as-query approach CRAWLDoc to determine relevant documents that encodes web content of various formats, anchor text, and URIs in a single embedding space.

- Evaluating document ranking (MRR, MAP, nDCG) and metadata extraction (BLEU, precision, recall) on 600 publications from the six largest computer science publishers,

4. `https://www.doi.org/`

demonstrating consistent improvement of CRAWLDoc over extracting only from the landing page.

- A robustness check to assess the generalizability of our system by training on five publishers and testing on a held-out publisher.
- A new dataset of bibliographic metadata with author affiliations, along with relevancy information for linked web documents. This dataset is the first of its kind to combine both document relevancy annotations and comprehensive bibliographic metadata.

Below, we summarize related work. We introduce our CRAWLDoc metadata extraction system in Section 3. The experimental apparatus is described in Section 4. The results are described in Section 5 and discussed in Section 6.

2 Related Work

We discuss research in neural information retrieval, retrieval-augmented and layout-aware language models, and scientific information extraction.

2.1 Neural Information Retrieval

Neural Information Retrieval (NIR) is a prominent research area, utilizing neural networks to improve the retrieval process. The landscape of NIR research has been extensively surveyed (Zhu et al., 2023; Zhang et al., 2016; Guo et al., 2020), highlighting the use of learned representations of queries and documents, commonly referred to as embeddings. These embeddings capture semantic similarities that traditional information retrieval models might overlook (Zhang et al., 2016; Mitra and Craswell, 2018; Abbasiantaeb and Momtazi, 2021). A pioneering model in this domain is the Deep Structured Semantic Model (DSSM) (Huang et al., 2013). DSSM is a latent semantic model that employs a deep neural network to project queries and documents into a common low-dimensional space. In this space, the relevance of a document to a query is determined by the distance between their respective projections. The BERT model (Devlin et al., 2019), although not specifically designed for information retrieval, has profoundly impacted NIR (Tian and Wang, 2021; Wang et al., 2024b; Li and Gaussier, 2022). BERT-based models such as CEDR (Contextualized Embeddings for Document Ranking) (MacAvaney et al., 2019) have achieved impressive performance on various information retrieval benchmarks. The ColBERT model (Khattab and Zaharia, 2020) introduced a late interaction paradigm, enabling efficient and effective passage retrieval. ColBERT's ability to balance effectiveness and efficiency has made it a popular choice for large-scale retrieval tasks (Santhanam et al., 2022).

Most NIR research focuses on query-to-document retrieval, where short queries are matched to documents. Document-to-document retrieval, where a full document serves as the query, is less common. PARM (Althammer et al., 2022) addresses this scenario for patent argument mining: it divides the query document into paragraphs, retrieves relevant paragraphs from a document corpus, and aggregates these paragraph-level similarities to produce a document-level ranking. This paragraph-aggregation approach is well-suited for long documents like patents but may be unnecessary for shorter documents such as publication landing pages.

2.2 Retrieval Augmentation

Pre-trained language models fine-tuned on downstream NLP tasks can store factual knowledge in their parameters and produce state-of-the-art results (Lewis et al., 2020; Fan et al., 2024). However, they perform less well on tasks requiring a high degree of knowledge. Retrieval augmentation aims to address this by augmenting the model's input with relevant information retrieved from external sources (Ram et al., 2023; Fan et al., 2024). REALM (Guu et al., 2020) is an early work that augments language models with a latent knowledge retriever. Based on the pre-training text, it allows the model to retrieve and attend over documents from a large corpus. REALM is pre-trained using masked language modeling as the learning signal and backpropagating through a retrieval step. Atlas (Izacard et al., 2023) is a model that can learn knowledge-intensive tasks with very few training examples. It uses retrieval during both pre-training and fine-tuning. RAG (Lewis et al., 2020) is a general-purpose fine-tuning method for retrieval-augmented generation. It combines pre-trained parametric and non-parametric memory for language generation. The parametric memory is a pre-trained Large Language Model (LLM) and the non-parametric memory is a dense vector index of a knowledge base, accessed with a neural retriever to find the top documents. RA-DIT (Lin et al., 2024) trains the retriever and the language model at the same time with two distinct fine-tuning steps: one updates a pre-trained language model to better use retrieved information, while the other updates the retriever to return more relevant results, as preferred by the LLM. By fine-tuning over tasks that require both knowledge utilization and contextual awareness, each stage yields substantial task performance improvements, and using both leads to additional gains.

CRAWLDoc can be viewed as a constrained form of retrieval-augmented generation where the retrieval corpus is limited to documents linked from a publication's landing page rather than a general knowledge base. This constraint is task-appropriate, as bibliographic metadata is inherently localized to the publication's web presence. Unlike standard RAG systems that retrieve from large corpora using keyword or semantic queries, CRAWLDoc uses the full landing page HTML as the query document to identify related pages that may contain complementary metadata.

2.3 Layout-Infused Language Models

Layout-infused language models consider both textual content and spatial layout. LayoutLMv3 (Huang et al., 2022) exemplifies this concept by pre-training multimodal transformers with a unified text and image masking objective, enhancing performance on both text-centric and image-centric tasks. Another approach is DocLLM (Wang et al., 2024a), which does not rely on expensive image encoders but relies solely on bounding box information from optical character recognition (OCR). This model is particularly useful for documents with irregular layouts and heterogeneous content. LMDX (Perot et al., 2023) is a model-agnostic method to adapt arbitrary LLMs for document information extraction. It extracts text with OCR and enriches it with layout information in the form of bounding boxes. The model proposes an XML-style prompt for information extraction and trains a text-only LLM with text and bounding boxes. The response of the LLM is decoded as a post-processing step based on the text and bounding boxes to discard hallucinations. Experiments show that LMDX is effective, especially in low data regimes. Layout-infused LLMs can face

challenges with layout distribution shifts. Chen et al. (2023) note that model performance can degrade by up to 20 points in macro F1 score under layout distribution shifts.

2.4 Scientific Information Extraction

Several academic metadata systems exist for indexing and searching scientific literature, including Google Scholar[5], OpenAlex[6], and Semantic Scholar[7]. Notably, affiliation extraction remains challenging, as even large-scale systems can associate many institutions with a single author profile, illustrating the difficulty of accurate per-paper affiliation assignment.

A seminal work on metadata collection and document-level metadata extraction is the CiteSeerX project (Li et al., 2006). Key to the system is a crawler for papers from authors' personal websites and other web sources. It employs a rule-based approach to detect affiliations, complemented by an SVM-based classifier that extracts metadata from the header of research papers (Han et al., 2003). Enlil (Do et al., 2013) addressed affiliation extraction through a pipeline that analyzes scholarly documents to extract and match authors and affiliations. While powerful, both CiteSeerX and Enlil focus on processing a single PDF document per paper and tend to rely on fixed approaches tailored to PDF structures. This single-document limitation makes them less applicable to heterogeneous multi-document scenarios that also involve HTML. Meanwhile, tools like CERMINE (Tkaczyk et al., 2015) and GROBID (Lopez, 2009) have also been utilized for scientific information extraction. Although these systems remain useful for extracting basic metadata from a single PDF, they do not address the multi-document aspect of handling heterogeneous web sources. Beyond these methods, commercial tools such as Elicit[8] offer standard solutions but similarly do not tackle multi-document metadata consolidation.

Beyond these earlier systems, the extraction of information from scientific text has, like many others, benefited from the advent of LLMs where they have demonstrated the ability to produce structured information from unstructured text (Xu et al., 2023). HyperPIE (Saier et al., 2023) is an approach for extracting hyperparameters from a scientific paper. It employs zero-shot generative models to generate YAML files incorporating the extracted hierarchical data.In the field of virology (Shamsabadi et al., 2024), LLMs have been employed for information extraction, showcasing the potential in domain-specific applications. A zero-shot GPT-3.5 (Brown et al., 2020) is compared to an instruction-tuned Flan-T5 (Chung et al., 2022) demonstrating the advantages of instruction-tuning the model. The versatility of LLMs is further exemplified in the medical field, where models like GPT-3 (Brown et al., 2020) have been applied for zero-shot and few-shot extraction of critical variables from clinical notes (Agrawal et al., 2022).

3 CRAWLDoc Metadata Extraction

We introduce CRAWLDoc (Contextual RAnking of Web-Linked Documents), a novel system to augment language models for entity extraction from multiple web sources. CRAWLDoc uses embeddings to identify relevant linked documents and then applies the results to

5. https://scholar.google.com
6. https://openalex.org
7. https://www.semanticscholar.org
8. https://elicit.com

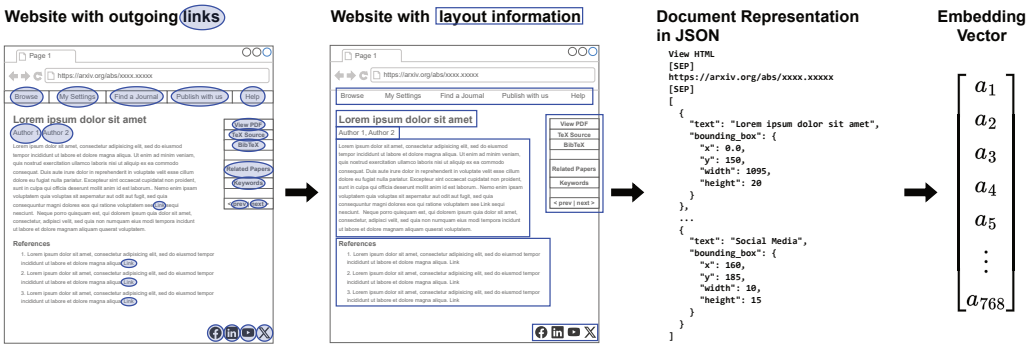


Figure 2: This figure illustrates the comprehensive process of our document representation methodology. The process begins with identifying all hyperlinks on the landing page, followed by integrating layout information by capturing bounding boxes. The document is then converted into a uniform textual format, which is finally encoded into a vector representation.

the extraction task. Specifically, the task is to identify bibliographic metadata of the entity classes: title, author names, author affiliations, and publication year.

Based on a seed URI, a DOI of a publication, CRAWLDoc scrapes linked resources, described in Section 3.1. Subsequently, the retrieved web documents in the form of HTML or PDF are ranked using a Small Language Model (SLM) (Lu et al., 2024). This is described in Section 3.2. The top k documents are passed on for information extraction, i. e., detecting and extracting relevant entities, using a LLM as described in Section 3.3. Our primary assumption for bibliographic metadata extraction is that all necessary information can be found within a one-hop crawl of the landing page associated with the DOI. This assumption is based on our observation that publishers present key bibliographic information on the landing page or pages directly linked to it e. g., the PDF of the publication.

3.1 Web Scraping from Seed URI

The initial stage of our system involves web scraping, starting with a DOI as the input and progressing to the scraping of the corresponding web page. After this starting point, all documents linked from the seed URI are retrieved, which may be formatted in HTML or PDF. Both PDF and HTML files undergo a series of steps to extract the relevant text and its associated bounding boxes to also capture layout information. For PDF documents, the text and its corresponding bounding box coordinates are directly extracted from the file using the PDFMiner Python library. In the case of HTML documents, the page is first rendered in a Firefox web browser (Version: 129.0.2) to accurately present the content's formatting and layout, and then the text and bounding boxes are extracted. This ensures that the textual content and its spatial context are preserved. This information is then converted into a uniform textual JSON format, which includes both the extracted text and its associated bounding box coordinates. This JSON representation serves as the input for both the ranking step and the subsequent extraction step, where the bounding box information

helps the LLM understand the document layout. This approach is inspired by layout-infused language models such as DocLLM (Wang et al., 2024a) and LMDX (Perot et al., 2023), which demonstrated that bounding box coordinates can effectively convey layout information to text-only models without requiring expensive image encoders. Figure 2 illustrates the different steps to create our document representation.

3.2 Neural Document Ranking

In the second step, we employ a SLM for the neural document ranking task. This task involves creating unified embeddings of the documents along with their associated anchor texts and URLs. For each document, we construct a single input representation by concatenating the anchor text, URL, and document content using a special separator token ([SEP]). This representation is then embedded into a dense vector space. The connection to Section 3.1 is illustrated in Figure 2. The document originating from the DOI (the landing page after rendering) is embedded utilizing a query encoder, and all documents linked from the landing page are embedded with the document encoder. A Maximum Inner Product Search (MIPS) is performed with the embedding of the landing page and the embeddings of all scraped documents to create a Contextual RAnking of Web-Linked Documents (CRAWLDoc) based on the landing page.

Unlike PARM (Althammer et al., 2022), which aggregates paragraph-level similarities for long patent documents, we embed the entire landing page as a single query vector. This simpler approach is sufficient for our setting, as publication landing pages are relatively short documents where paragraph-level decomposition would provide little benefit.

We use the jina-embedding-2 model (Günther et al., 2023) as neural retriever. Following the success of BERT-based models in neural information retrieval (Wang et al., 2024b), Jina-v2 is based on a BERT (Devlin et al., 2019) architecture and supports the symmetric bidirectional variant of ALiBi (Press et al., 2022), allowing for a sequence length of up to 81,921 tokens. Due to memory restrictions, we limit our experiments to the first 2,048 tokens. The neural retriever is trained using contrastive learning with the InfoNCE loss function (van den Oord et al., 2018).

3.3 Information Extraction

Following the retrieval-augmented generation (RAG) paradigm (Lewis et al., 2020), we use the retrieved and ranked documents as context for a language model that performs the extraction. To facilitate the extraction of bibliographic metadata, we employ XML-style prompts that instruct large language models to extract the required metadata from the document and respond in a specific JSON format. Detailed information on the prompt design can be found in Appendix C.

The process begins with the ranking scores obtained from the previous step. These scores are used to concatenate the documents in the order of relevance, placing the most relevant documents first. This follows common retrieval-augmented generation practice of providing retrieved context in retriever-score order (Lewis et al., 2020) and is further motivated by evidence that language models attend more strongly to information at the beginning of their context window (Liu et al., 2024). To optimize the process, we limit the number of documents to a maximum of the top five ($k = 5$) ranked documents. This

decision is based on our observation that the average number of relevant documents per publication in our dataset is 5.45 (see Section 4.1). The concatenated documents serve as context for the LLM, which extracts key bibliographic metadata. The final output is a JSON object containing the title, author name, author affiliation, and publication year. This multi-document approach distinguishes CRAWLDoc from single-document extractors like GROBID (Lopez, 2009) and CERMINE (Tkaczyk et al., 2015), enabling the system to aggregate information scattered across multiple sources.

For information extraction, we use the GPT-4o model *gpt-4o-2024-08-06* via API. This model is particularly suited due to its extensive context window, which accommodates up to 128,000 tokens, allowing us to process large volumes of text efficiently. The system outputs the extracted information in a structured JSON format, allowing for straightforward parsing and integration.

4 Experimental Apparatus

4.1 Dataset

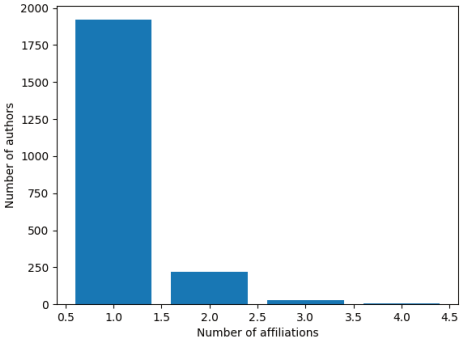
We use the DBLP Computer Science Bibliography dataset.[9] We take a subset of bibliographies from the six largest publishers in computer science, which together represent more than 80% of all publications listed in DBLP (see Appendix A for the full publisher distribution). This ensures the dataset contains a representative set of layouts encountered for bibliographic metadata extraction. We randomly select 100 publications for each publisher and split them into training, validation, and test sets in an 80/10/10 ratio, ensuring each publisher has the same number of publications per split.

Dataset Annotation We obtained the metadata for each publication by manually retrieving the title, publication year, and authors' names and affiliations. We retrieved the landing page of each publication and labeled every outgoing link on the landing page with a binary relevancy label. A linked document is labeled as *relevant* if it refers to the same publication and contains any of our target metadata fields (title, authors, affiliations, or publication year). Otherwise, it is labeled as *not relevant*. Typical examples of relevant documents include the publication PDF itself, author profile pages (e. g., ORCID), institutional author pages, associated GitHub repositories, and linked BibTeX files. The annotation was performed by a single expert annotator (the first author). By manually creating this dataset, we ensure high quality of the metadata and can accurately assess the document retrieval process in our proposed setup. The tool for conducting the labeling is documented in Appendix D.

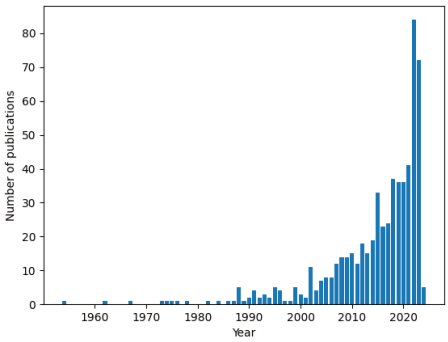
To prevent artificial inflation of our performance metrics, we identified and removed any instances in our test set where the landing page contained links to itself (e. g., through self-referential navigation elements). The trivial nature of calculating document similarity to itself would otherwise result in an unrepresentative boost in ranking performance.

Dataset Statistics Our dataset consists of 600 publications with detailed metadata and 72,483 linked documents with binary relevancy labels. Per publication, we have on average of 3.63 (SD: 2.10) authors, with an average of 1.14 (SD: 0.41) affiliations per author. Figure 3

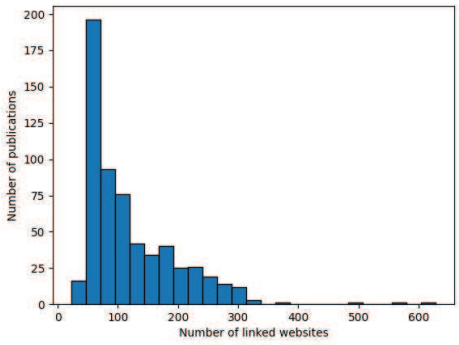
9. `https://dblp.org/xml/release/dblp-2024-04-01.xml.gz`



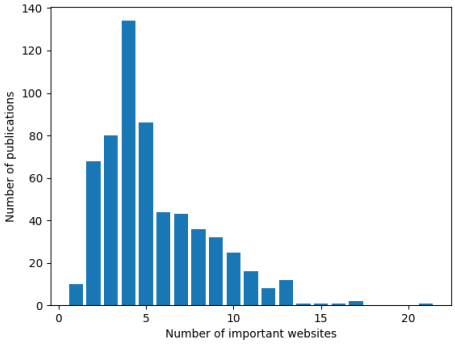
(a) Affiliations per author



(b) Publications per publication year



(c) Linked documents per publication



(d) Relevant labeled documents per publication

Figure 3: The figure presents several aspects of the dataset: (a) the number of affiliations associated with each author, (b) the yearly distribution of publications, (c) the count of linked documents per publication, and (d) the number of documents per publication that have been assigned a positive relevancy label. Each subfigure illustrates these respective distributions.

provides visualizations of important characteristics of our dataset. Most authors have a single affiliation, but some have up to four affiliations. Our dataset includes publications from a wide range of years, with a higher proportion of recent publications due to the larger volume of newer publications listed on DBLP. On average, each publication has 120.81 (SD: 76.52) linked documents, but only 5.45 (SD: 2.99) of these documents are relevant to the publication. The high number of linked documents is due to the inclusion of all hyperlinks found on the landing page, including navigation menus, footer links, related article suggestions, and publisher-wide resources, most of which are not relevant to the specific publication.

To the best of our knowledge, we are the first to release a dataset that includes author affiliations as mentioned in the publications. Additionally, we are the first to provide

relevancy labels for linked documents in the context of publication web data. For legal purposes, we are only able to publish the labels and not the actual website content. However, we publish the landing page URLs along with the relevancy labels, and provide the source code[10] to reproduce the data collection procedure.

Dataset License The DBLP dataset is released under CC0 1.0 Public Domain Dedication license. Our annotations have the same license.

4.2 Procedure

Our experimental procedure for document ranking involves fine-tuning a neural document retriever using contrastive loss to improve document ranking. To ensure robust performance, we evaluate the ranking capabilities on both in-distribution and out-of-distribution data. For information extraction from the selected sources, we employ a zero-shot language model (GPT-4o) to extract bibliographic metadata, assessing the entire system's effectiveness. To demonstrate the advantages of our CRAWLDoc system, we compare it with a baseline setup that uses only the publication's landing page.

4.3 Hyperparameter Optimization

We optimize the hyperparameters of the neural document retriever to maximize retrieval performance on the validation set, using nDCG as the selection criterion. Specifically, we set the batch size to two queries and five negative examples, which is the maximum that fits on a single H100 with 80GB of VRAM. Since contrastive learning benefits greatly from a large batch size (Chen et al., 2022), we further optimize the number of accumulation steps. The grid search for hyperparameters includes the following values: learning rate (1e-05, 2e-05, 3e-05), number of accumulation steps (1, 16, 32, 64), and patience (2, 5). We train the model for up to 25 epochs with early stopping and optimize the early stopping patience. The optimized hyperparameters are a learning rate of 3e-05, 32 accumulation steps, and patience of 5, resulting in 16 epochs. The hyperparameter optimization was performed using all six publishers in the training set. For the leave-one-out experiments, the hyperparameters were not re-tuned for the reduced training set.

For metadata extraction, we set the number of documents used as context to $k = 5$. This value balances the need for comprehensive information gathering with computational efficiency. By limiting the number of documents, we ensure a focused set of highly relevant documents for the extraction process while maintaining a manageable computational load. The decision to set $k = 5$ is based on our observation that the average number of relevant documents per landing page is 5.45, as discussed in Section 4.1.

4.4 Metrics

To evaluate the ranking of the web documents, we employ several metrics. The Mean Reciprocal Rank (MRR) evaluates the effectiveness of a retrieval system by considering the rank position of the first relevant result. It is calculated as the average of the reciprocal ranks of the first relevant result across a set of queries. Formally, MRR is defined as:

10. `https://github.com/FKarl/crawldoc-metadata-extraction`

Publisher	Ranking			Extraction		
	MRR	**MAP**	**nDCG**	**BLEU**	**P**	**R**
IEEE	1.000	1.000	1.000	0.901	1.000	0.937
Springer	0.800	0.998	0.800	0.913	1.000	0.944
Elsevier	1.000	0.970	0.985	0.962	0.991	0.976
ACM	1.000	0.999	1.000	0.712	0.872	0.773
arXiv	1.000	1.000	1.000	0.902	1.000	0.972
MDPI	1.000	0.954	0.982	0.954	1.000	0.979
All	0.967	0.987	0.961	0.891	0.977	0.930

Table 1: Performance metrics for the ranking and extraction tasks across different publishers. 'P' denotes n-gram precision and 'R' is n-gram recall. Values are provided for each publisher, along with aggregated results for all publishers. For the extraction task, the macro average is reported.

MRR $= \frac{1}{|Q|}\sum_{i=1,\ldots,|Q|}\frac{1}{rank_i}$. The MRR focuses on the first relevant document in the ranked list, i. e., it favors a relevant document in the highest position. In contrast, Mean Average Precision (MAP) evaluates the precision of a retrieval system by averaging the precision scores at all ranks where relevant documents are found and then averaging these scores over all queries. Normalized Discounted Cumulative Gain (nDCG) (Järvelin and Kekäläinen, 2002) measures the usefulness of a document based on its position in the result list, assuming that highly relevant documents are more useful when appearing earlier. It is computed by normalizing the Discounted Cumulative Gain (DCG) by the ideal DCG (iDCG). Since our relevancy labels are binary (relevant/not relevant), DCG uses gains of 1 for relevant documents and 0 for non-relevant documents. We compute nDCG over the full ranking without a cutoff. We further calculate the precision@k, recall@k, and F1@k which measure the proportion of relevant items in the top k results.

For the extraction task, we employ the BLEU score
citepBLEU, a metric commonly used in machine translation to evaluate the similarity between the generated output and the reference. Given the nature of our generated outputs, which tend to be concise, we focus on word-level unigram BLEU scores, as higher-order n-grams (such as 2 to 4 grams) may not consistently occur in shorter text segments. To provide a more granular analysis, we also calculate n-gram precision and recall at the character level, considering 1 to 4 grams, following standard practice in text generation evaluation citepBLEU.

5 Results

5.1 Document Ranking

The ranking results for identifying relevant linked documents are shown in Table 1. Overall, we achieve an average ranking performance of MRR 0.967, MAP 0.987, and nDCG 0.961. The MRR, MAP, and nDCG values exhibit a consistently high level of performance for the six publishers, except for MRR and nDCG on the Springer dataset. The MRR for IEEE,

Method	MRR	MAP	nDCG
Jina-v2 (zero-shot)	0.045	0.094	0.280
BM25	0.387	0.244	0.451
BM25+	0.399	0.268	0.477
Jina-v2 (fine-tuned, ours)	**0.967**	**0.987**	**0.961**

Table 2: Comparison of document ranking performance across different retrieval methods. Our fine-tuned neural retriever substantially outperforms both sparse retrieval baselines (BM25, BM25+) and the zero-shot dense retriever. Results are averaged across all six publishers.

Elsevier, ACM, arXiv, and MDPI all achieve the maximum score of 1.000, indicating that a relevant document is always in the top position.

Comparison with Baselines To demonstrate the effectiveness of our neural document ranking approach, we compare it against several baselines. These include BM25 (Robertson et al., 1994), a widely used sparse retrieval method, BM25+ (Lv and Zhai, 2011), which addresses BM25's over-penalization of long documents by lower-bounding each term's contribution, and Jina-v2 (zero-shot), the same embedding model we use but without any fine-tuning on our dataset. Table 2 presents this comparison.

The fine-tuned neural retriever substantially outperforms all baselines across all metrics. Compared to the best sparse baseline (BM25+), our approach achieves an MRR of 0.967 versus 0.399. Similarly, nDCG improves from 0.477 to 0.961 and MAP from 0.268 to 0.987. A detailed per-publisher breakdown is provided in Appendix B. The improvement is consistent across all publishers, with particularly large gains for Springer (MRR from 0.178 to 0.800) and ACM (MRR from 0.207 to 1.000).

Notably, the zero-shot Jina-v2 embeddings perform substantially worse than even BM25 (MRR of 0.045 versus 0.387), demonstrating that fine-tuning on domain-specific data is essential for this task. The poor zero-shot performance can be attributed to the unique characteristics of our document-as-query setting, where the model must identify documents about the same publication rather than semantically similar documents in general. These results confirm that our embedding-based approach, when properly fine-tuned, effectively handles the challenge of diverse web sources with varying structures and formats.

To understand the impact of layout information on ranking performance, we conducted an ablation study. The results without layout information showed slightly lower performance with an MRR of 0.950, a MAP of 0.976, and an nDCG of 0.952.

We have conducted a more detailed examination of the ranking performance with different cut-off values k visualized in Figure 4. The recall increases with increasing values of k, reaching 0.97 at $k = 20$. Precision declines from 0.97 for $k = 1$ to 0.22 for $k = 20$. The F1@k score, which combines precision and recall, reaches its highest value of 0.77 for $k = 4$ and $k = 5$.

We have evaluated robustness using a leave-one-out strategy, training on all but one publisher and testing on the left-out publisher. The results of the robustness analysis are shown in Table 3. We obtain a high performance across all publishers, with an average

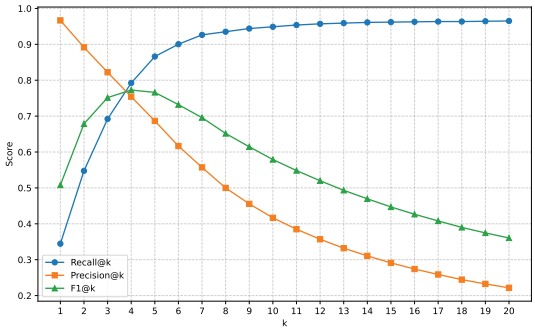


Figure 4: Visualization of the ranking performance evaluation of the model at different cut-off values k.

Tested on	MRR	MAP	nDCG
IEEE	1.000	1.000	1.000
Springer	0.757	0.835	0.772
Elsevier	1.000	0.996	0.999
ACM	1.000	0.999	1.000
arXiv	1.000	1.000	1.000
MDPI	1.000	0.979	0.992
Average	0.959	0.968	0.961

Table 3: Performance of the ranking task across all publishers in a leave-one-out test. The publisher the model is "tested on" is not part of the training data. The results are provided per publisher, along with the average performance across all publishers.

MRR of 0.959, MAP of 0.968, and nDCG of 0.961. This is less than one point for MRR and nDCG and less than two points for MAP compared to using the full training dataset shown in Table 1. For IEEE and arXiv, the model achieves the maximum score of 1.000 for all three metrics. However, the performance was slightly lower for Springer, consistent with the result on the full training set.

5.2 Information Extraction

The extraction performance of our system is assessed by calculating BLEU scores, n-gram precision (P), and n-gram recall (R). The results can be found in the right column of Table 1. CRAWLDoc achieves an overall BLEU score of 0.891, precision of 0.977, and recall of 0.930 across all publishers. IEEE, Springer, arXiv, and MDPI achieved perfect precision scores of 1.000, with Elsevier at 0.991. Recall ranges from 0.937 to 0.979 for these publishers. In contrast, ACM showed the lowest performance with a BLEU score of 0.712, precision of 0.872, and recall of 0.773.

Table 4 shows the extraction performance per entity type, i. e., title, author name, author affiliations, and publication year. The extraction of titles and publication years from all publishers achieved near-perfect accuracy, with average BLEU scores of 0.947 and 0.950, a

Publisher	Title			Author Names			Author Affiliations			Publication Year		
	BLEU	**P**	**R**	**BLEU**	**P**	**R**	**BLEU**	**P**	**R**	**BLEU**	**P**	**R**
IEEE	1.000	1.000	1.000	0.784	1.000	0.884	0.821	1.000	0.863	1.000	1.000	1.000
Springer	0.925	1.000	0.946	0.875	1.000	0.899	0.955	1.000	0.971	0.900	1.000	0.960
Elsevier	0.989	1.000	0.992	0.926	1.000	0.975	0.932	0.963	0.938	1.000	1.000	1.000
ACM	0.946	1.000	1.000	0.670	0.838	0.710	0.231	0.649	0.381	1.000	1.000	1.000
arXiv	0.909	1.000	1.000	0.961	1.000	0.977	0.940	1.000	0.987	0.800	1.000	0.925
MDPI	0.914	1.000	0.936	0.972	1.000	0.994	0.932	1.000	0.987	1.000	1.000	1.000
All	0.947	1.000	0.979	0.876	0.974	0.913	0.810	0.939	0.865	0.950	1.000	0.981

Table 4: Extraction performance with GPT-4o (OpenAI, 2023), given the top-5 documents selected by CRAWLDoc, for the different entity types. 'P' denotes n-gram precision and 'R' is n-gram recall. Metrics are provided per publisher as well as aggregated over all.

Publisher	Names & Affiliations		
	BLEU	**P**	**R**
IEEE	0.613	1.000	0.941
Springer	0.654	1.000	0.984
Elsevier	0.710	1.000	0.957
ACM	0.150	1.000	0.569
arXiv	0.617	1.000	0.978
MDPI	0.494	1.000	0.981
All	0.540	1.000	0.902

Table 5: Breakdown of the extraction task performance for treating author names and affiliation as one entity without hierarchical relation. 'P' denotes n-gram precision and 'R' is n-gram recall. Metrics are provided per publisher as well as aggregated over all publishers.

recall of 0.979 and 0.981, and a precision of 1.000 for both entity types, respectively. Author names and affiliations were more difficult to extract. For affiliation extraction, a notable decline in performance is observed for the ACM dataset with a BLEU score of 0.231 and recall of 0.381. We observe a very low standard deviation, except for ACM (details are provided in Appendix E).

In addition, we conducted further analysis to assess the impact of ignoring the hierarchical relationship between authors, their names, and their affiliations. Specifically, we treated author names and affiliations as a single entity during extraction, instead of considering their natural hierarchy. The results, shown in Table 5, indicate that CRAWLDoc achieves a BLEU score of 0.540, precision of 1.000, and recall of 0.902 across all publishers.

In addition to the multi-document setup of CRAWLDoc, we report in Tables 6 to 8 experimental results where GROBID, CERMINE, and GPT-4o each received only the correct single PDF for extraction. In other words, instead of considering the multi-document case, we analyze the effectiveness of three metadata extractors here. We assume that the ranking optimally identified the top-ranked document as the correct one and passed it on to the extractor.

Both GROBID and CERMINE (Tables 6 and 7) achieve higher scores for extracting titles and publication years than for affiliations, and affiliation metrics vary across publishers.

Publisher	Title			Author Names			Author Affiliations			Publication Year		
	BLEU	P	R	BLEU	P	R	BLEU	P	R	BLEU	P	R
IEEE	0.395	0.800	0.788	0.418	0.724	0.614	0.022	0.448	0.128	0.000	0.000	0.000
Springer	0.857	0.857	0.857	0.525	1.000	0.640	0.042	0.517	0.165	0.047	0.142	0.142
Elsevier	0.807	1.000	0.983	0.820	1.000	0.891	0.117	0.769	0.295	0.111	0.555	0.461
ACM	0.649	1.000	1.000	0.797	0.941	0.855	0.180	0.705	0.357	0.000	0.000	0.000
arXiv	0.883	1.000	1.000	0.846	0.983	0.877	0.099	0.610	0.252	0.166	0.600	0.562
MDPI	0.990	1.000	0.998	0.930	1.000	0.971	0.061	1.000	0.266	0.333	1.000	1.000
All	0.760	0.945	0.940	0.752	0.949	0.827	0.089	0.684	0.248	0.115	0.400	0.377

Table 6: Extraction performance with GROBID (Lopez, 2009), always provided with the perfect PDF for extraction, for the different entity types. 'P' denotes n-gram precision and 'R' is n-gram recall. Metrics are provided per publisher as well as aggregated over all.

Publisher	Title			Author Names			Author Affiliations			Publication Year		
	BLEU	P	R	BLEU	P	R	BLEU	P	R	BLEU	P	R
IEEE	0.612	1.000	0.928	0.494	0.862	0.624	0.247	0.482	0.334	0.700	0.900	0.792
Springer	0.722	1.000	0.745	0.534	0.793	0.616	0.314	0.586	0.456	0.714	1.000	0.817
Elsevier	0.618	1.000	0.724	0.800	1.000	0.871	0.570	1.000	0.751	0.666	1.000	0.856
ACM	0.575	0.888	0.792	0.494	0.647	0.558	0.242	0.617	0.356	0.777	0.888	0.839
arXiv	0.883	1.000	1.000	0.452	0.762	0.623	0.298	0.745	0.464	0.000	0.300	0.130
MDPI	1.000	1.000	1.000	0.916	0.976	0.959	0.273	0.523	0.371	1.000	1.000	1.000
All	0.741	0.981	0.875	0.605	0.831	0.706	0.312	0.657	0.445	0.636	0.836	0.731

Table 7: Extraction performance with CERMINE (Tkaczyk et al., 2015), always provided with the perfect PDF for extraction, for the different entity types. 'P' denotes n-gram precision and 'R' is n-gram recall. Metrics are provided per publisher as well as aggregated over all.

Table 8 shows that GPT-4o with only one PDF exhibits a similar pattern, with higher scores for titles and publication years than for affiliations. Depending on the publisher, affiliation and author-name extraction can be higher or lower compared to GROBID and CERMINE.

A comparison of the average results for these single document extractors, along with CRAWLDoc, can be seen in Table 9. Notably, CRAWLDoc matches or exceeds all three single-document extractors (GPT-4o, GROBID, and CERMINE) across all evaluated entity types, despite not being provided with the perfect PDF.

We also compare CRAWLDoc, which augments the landing page with additional web content, to a baseline using only the landing page. The results of this experiment can be seen in Table 10.

To set these results in relation, Table 11 compares the averaged extraction performance of the landing page only setup and CRAWLDoc. In this comparison CRAWLDoc consistently outperforms the baseline for all six publishers. On average, our retrieval-augmented CRAWLDoc achieves a BLEU score of 0.891, compared to 0.745 for the landing page extraction. Precision and recall were also higher for CRAWLDoc, with improvements most noticeable for the publishers ACM and arXiv. For example, the BLEU score for arXiv jumped from 0.469 for landing-page-only extraction to 0.902 with CRAWLDoc, highlight-

Publisher	Title			Author Names			Author Affiliations			Publication Year		
	BLEU	P	R	BLEU	P	R	BLEU	P	R	BLEU	P	R
IEEE	0.613	1.000	1.000	0.920	1.000	0.974	0.843	1.000	0.931	0.900	1.000	0.962
Springer	0.794	1.000	0.860	0.923	1.000	0.971	0.771	1.000	0.853	0.875	1.000	0.950
Elsevier	0.788	1.000	0.905	0.691	1.000	0.819	0.807	1.000	0.858	0.900	1.000	0.930
ACM	0.603	1.000	0.922	0.875	1.000	0.912	0.431	0.973	0.644	0.900	1.000	0.950
arXiv	0.869	1.000	0.998	0.884	1.000	0.909	0.921	1.000	0.979	0.800	1.000	0.925
MDPI	0.990	1.000	0.998	0.971	1.000	0.993	0.893	1.000	0.963	1.000	1.000	1.000
All	0.776	1.000	0.950	0.885	1.000	0.931	0.791	0.995	0.883	0.896	1.000	0.953

Table 8: Extraction performance with GPT-4o (OpenAI, 2023), always provided with the perfect PDF in the form of JSON for extraction, for the different entity types. 'P' denotes n-gram precision and 'R' is n-gram recall. Metrics are provided per publisher as well as aggregated over all.

Publisher	Title			Author Names			Author Affiliations			Publication Year		
	BLEU	P	R	BLEU	P	R	BLEU	P	R	BLEU	P	R
GPT-4o	0.776	1.000	0.950	0.885	1.000	0.931	0.791	0.995	0.883	0.896	1.000	0.953
GROBID	0.760	0.945	0.940	0.752	0.949	0.827	0.089	0.684	0.248	0.115	0.400	0.377
CERMINE	0.741	0.981	0.875	0.605	0.831	0.706	0.312	0.657	0.445	0.636	0.836	0.731
CRAWLDoc	0.947	1.000	0.979	0.876	0.974	0.913	0.810	0.939	0.865	0.950	1.000	0.981

Table 9: Extraction performance comparison of CRAWLDoc and baseline single-document extractors (GPT-4o, GROBID, CERMINE). Each baseline extractor is always provided with the perfect PDF, while CRAWLDoc operates in a multi-document setting without manual intervention. 'P' denotes n-gram precision and 'R' is n-gram recall. Metrics are provided as aggregated over all publishers.

ing the advantage of utilizing the CRAWLDoc system for more comprehensive and accurate data extraction.

5.3 Computational Cost

We report the computational cost of GPT-4o extraction to demonstrate practicability. Across 60 test samples (10 per publisher), the average input token count is 84,488 tokens per publication with an average of 178 output tokens. At current API pricing ($2.50 per million input tokens and $10.00 per million output tokens), the average cost per publication is $0.21. The total cost for processing all 60 test samples was $12.78. Table 12 presents a detailed breakdown per publisher. Costs vary due to differing document lengths, with MDPI averaging the highest cost at $0.31 per publication and ACM the lowest at $0.14.

6 Discussion

6.1 Key Scientific Insights

Document Ranking The outcomes of our research illustrate the effectiveness of our proposed document ranking and extraction system. Our system achieves high ranking performance, with relevant documents frequently appearing at the top ranks, which estab-

Publisher	Title			Author Names			Affiliations			Publication Year		
	BLEU	**P**	**R**	**BLEU**	**P**	**R**	**BLEU**	**P**	**R**	**BLEU**	**P**	**R**
IEEE	1.000	1.000	1.000	0.784	1.000	0.884	0.806	0.828	0.820	0.900	1.000	0.956
Springer	0.925	1.000	0.946	0.775	0.914	0.797	0.843	0.914	0.864	0.900	1.000	0.960
Elsevier	0.711	1.000	0.749	0.938	1.000	0.983	0.813	0.889	0.856	0.800	1.000	0.911
ACM	1.000	1.000	1.000	0.000	0.649	0.054	0.000	0.216	0.036	1.000	1.000	1.000
arXiv	0.009	1.000	0.121	0.969	1.000	0.981	0.000	0.000	0.000	0.900	1.000	0.944
MDPI	0.900	1.000	0.921	0.972	1.000	0.994	0.932	1.000	0.987	1.000	1.000	1.000
All	0.757	1.000	0.790	0.756	0.930	0.793	0.498	0.568	0.524	0.917	1.000	0.962

Table 10: Breakdown of the extraction task performance metrics for the different elements that are extracted in the landing page only setup. Note that 'P' denotes n-gram Precision, and 'R' represents n-gram Recall. Metrics are provided for each publisher individually, as well as aggregated results over all publishers.

Publisher	CRAWLDoc			Only the landing page		
	BLEU	**P**	**R**	**BLEU**	**P**	**R**
IEEE	0.901	1.000	0.937	0.872	0.957	0.915
Springer	0.913	1.000	0.944	0.861	0.957	0.892
Elsevier	0.962	0.991	0.976	0.816	0.972	0.875
ACM	0.712	0.872	0.773	0.500	0.716	0.523
arXiv	0.902	1.000	0.972	0.469	0.750	0.512
MDPI	0.954	1.000	0.979	0.951	1.000	0.976
All	0.891	0.977	0.930	0.745	0.892	0.782

Table 11: Extraction performance of CRAWLDoc compared to using only the landing page. 'P' denotes n-gram precision and 'R' represents n-gram recall. Values are provided for each publisher, along with aggregated results for all publishers. The macro average is reported to provide a comprehensive evaluation of the performance across different publishers.

lishes a strong basis for the subsequent extraction task. This trend is seen when examining the evaluation of ranking performance at various cutoff values. We notice a sharp rise in recall@k for the first few documents, but only minor enhancements after around five documents. The decline in precision@k as k values increase is a natural result considering that a publication has on average 5.45 relevant documents per publication (see Section 4.1). This is also reflected in the F1@k score, which is peaking at $k = 4$ and $k = 5$. Overall, the results show that CRAWLDoc maintains a good balance between precision and recall with a cut-off value of $k = 5$.

The comparison between CRAWLDoc and extraction solely from the landing page (Table 11) illustrates the advantages of our system. The improvement is especially noticeable in situations such as arXiv, where the landing page lacks affiliation data. This emphasizes the importance of utilizing information from linked documents.

Robustness of Document Ranking Our model demonstrates robust performance across different publishers within our evaluated scope. While previous research, such as Chen et al. (2023), has identified challenges for layout-infused LLMs when dealing with layout distribu-

Publisher	Input Tokens	Output Tokens	Cost ($/DOI)
IEEE	90,881	137	0.23
Springer	69,715	192	0.18
Elsevier	90,948	141	0.23
ACM	56,049	96	0.14
arXiv	75,794	273	0.19
MDPI	123,543	230	0.31
Average	84,488	178	0.21

Table 12: Computational cost of GPT-4o extraction per publisher. Input and output tokens are averaged across 10 test samples per publisher. Cost is calculated at $2.50 per million input tokens and $10.00 per million output tokens.

tion shifts, our system shows consistent performance. This is evidenced by nearly equivalent performance between in-distribution and out-of-distribution data, suggesting effective generalization. Academic publishers often follow similar design patterns and conventions for their publication pages, reducing the effective layout distribution shift between sources. This standardization is further reinforced by the widespread adoption of common publishing platforms, such as Open Journal Systems[11] among smaller publishers. Our robustness evaluation considered six major publishers. The conventional nature of academic publication layouts suggests that the approach may transfer to additional publishers, but further evaluation is required.

Information Extraction Our system achieves high extraction performance, but its recall could be improved. The lower recall can be attributed to a tendency for over-inclusion rather than hallucination during error occurrences. For instance, the model may retrieve a more detailed title including a subtitle or the conference name, rather than the more concise title explicitly stated in the publication. When assessing the extraction performance of author information at the entity level, our analysis reveals specific error patterns in entity counting. We have observed cases of author number mismatches (1 case of over-extraction, 3 cases of under-extraction) and affiliation mismatches (4 cases of over-extraction, 14 cases of under-extraction). These entity counting errors contribute to the decline in overall performance. Moreover, affiliation extraction is more difficult due to fewer documents that contain this information in comparison to other entities such as the title. The model also occasionally excludes components of the affiliation strings, such as ZIP codes, or provides abbreviated versions. The discrepancies could be caused by the differing levels of specificity presented across different documents. For the ACM dataset, we observe that author profile pages frequently appear among the top-ranked documents returned by CRAWLDoc's neural retriever. However, these pages often present a comprehensive history of an author's professional affiliations, including both past and present. A closer inspection of the extracted affiliations reveals that inaccuracies can be attributed to these profile pages where the model retrieves outdated or newer affiliations, rather than focusing on the affiliations relevant to the time of publication.

11. `https://pkp.sfu.ca/software/ojs/`

When confined to a single PDF (Tables 6–8), GROBID, CERMINE and GPT-4o achieve comparable results for titles and years, reflecting the relative simplicity of locating these fields. However, the limited context of a single PDF can hinder affiliation extraction because relevant details sometimes appear in multiple places (e. g., supplemental or author-profile links). Consequently, GPT-4o under CRAWLDoc (Table 4) gains an advantage by integrating additional, potentially clearer affiliation fields. Still, Tables 6–8 confirm that under optimal single-document conditions, these single-PDF extractors (GROBID, CERMINE, and GPT-4o with one PDF) can achieve comparable performance for some publishers and evaluation metrics.

Finally, an interesting observation from our analysis concerns the role of layout information in the extraction process. While the bounding box coordinates used in our pipeline are beneficial to the extraction, the impact on the performance is not strong. In other words, the LLM that operates on the JSON representation of extracted text can align the text fields well to their respective roles (title, author, affiliation, etc.) without knowing where the text is located in the paper.

Summary In summary, CRAWLDoc outperforms extraction from the landing page alone across all six publishers and all evaluation metrics (Table 11). For document ranking, the neural retriever achieves MRR of 0.967 and nDCG of 0.961, substantially outperforming the BM25 baseline (Table 2). For metadata extraction, CRAWLDoc achieves higher BLEU, precision, and recall than single-document extractors (GROBID, CERMINE, GPT-4o with one PDF) on average (Table 9), despite operating without knowledge of which document is the publication PDF. The leave-one-out experiment (Table 3) confirms generalization to unseen publisher layouts. However, specific publishers like Springer and ACM exhibit lower performance, indicating room for improvement.

6.2 Generalization and Threat to Validity

The generalizability of our work refers to different publishers based on a leave-one-out test (Table 3) in which we test the system for publishers on which it has not been trained. Our robustness check demonstrates that a trained model can achieve comparably good results on out-of-distribution data within our tested scope. This finding suggests that our model has learned generalizable features of document relevance and metadata extraction that extend beyond the specific layouts and publishers in our training data. Moreover, our approach of transforming different document formats (HTML and PDF) into a uniform textual representation enhances its potential for generalization. This uniformity in representation suggests applicability to other web and document-related tasks beyond bibliographic metadata extraction. The ability to handle diverse document formats while maintaining high performance is a valuable characteristic that could facilitate the application of our system to other data processing tasks, which are outlined below.

While our study provides robust results, it is important to reflect on potential threats to validity.

Publisher Coverage One threat is the limited scope of our investigation, which focuses on only six publishers, primarily from the computer science field. These publishers represent the “fat head” of scientific publishing, together accounting for more than 80% of publications

in computer science (see Appendix A). While this provides a representative set of layouts for high-volume publishers, the remaining 20% of publications (the "long tail") may exhibit greater variability in document layouts and metadata presentation. Future work should evaluate CRAWLDoc on smaller publishers to assess generalization to this long tail.

Component Optimization Another consideration is the extent to which system components were optimized across all publishers. The extraction prompt (Appendix C) was designed using publications from all six publishers, meaning the leave-one-out experiment does not fully isolate the held-out publisher. However, the prompt contains no publisher-specific instructions and was not refined for individual publishers. Similarly, hyperparameters for the neural retriever were tuned on the full training set and not re-optimized for the leave-one-out experiments. These design choices reflect a practical deployment scenario where the system is configured once and applied to new publishers without re-tuning.

Recency Bias An additional possible risk is the presence of recency bias in our dataset, given that the majority of publications in DBLP are from more recent years. Nevertheless, we have found that older publications, including papers as far back as 1967, in our test set achieve similar performance to more recent ones, which eases this concern. This indicates that the performance of our model is not much influenced by the publication year.

In terms of speed, our method depends on the speed of the web crawling, the efficiency of embedding the documents, and the maximum inner product search (MIPS). As embedding model, we rely on `jina-embeddings-v2`, which contains only 137 million parameters (Günther et al., 2023). A landing page has, on average, 120.81 linked documents. Once the DOI is resolved, the crawling of those documents is efficient.

Moreover, our approach is easily parallelizable as each paper is handled independently. While there is no particular reason why not other embedding models could be used, too, our work does not focus on finding optimal embedding models for the retrieval tasks. We use Jina embeddings because they are widely used and have demonstrated strong results (Günther et al., 2023).

6.3 Future Work and Impact

Rerankers (Zhu et al., 2023) could yield additional improvements in document ranking accuracy. Future work could also explore alternative neural retriever setups like ColBERTv2 (Santhanam et al., 2022) and token-level representation of documents with MaxSim (Khattab and Zaharia, 2020) instead of cosine similarity.

Approaches like Enlil (Do et al., 2013) and others could be adapted and used in the information extraction step of CRAWLDoc. In such a scenario, each ranked document would be processed independently rather than concatenated. Similarly, services like CiteSeerX (Li et al., 2006) could incorporate CRAWLDoc as a component for multi-source and multi-format bibliographic metadata extraction.

For instance, a service like CiteSeerX could use CRAWLDoc in the future to identify affiliation information. Since CiteSeerX has a general-purpose web crawler, it could pass a URI to our service, and we would return affiliation information along with standard bibliographic metadata.

Instruction-tuning a language model to utilize layout information better, as demonstrated by Perot et al. (2023), is another direction worth exploring for documents with complex or varied layouts. Developing methods to identify important sections within documents could optimize context utilization, potentially improving the efficiency and accuracy of our extraction process. This could be particularly beneficial when dealing with long documents or when processing time is a constraint. Our system can also be valuable in legal and patent search (Nguyen et al., 2024) for retrieving and extracting precise information from diverse documents.

7 Conclusion

Our Contextual RAnking of Web-Linked Documents (CRAWLDoc) retrieval system is effective in improving a language model's ability to extract bibliographic metadata from various web sources. The key scientific findings include the effective identification of relevant web documents to improve the performance of the information extraction task and the robustness of the system across different publishers and their web-layout. The retrieval-augmented CRAWLDoc system consistently yields better extractions than relying only on the landing page. This supports the idea that linked documents can provide useful extra context, especially when the landing page does not have enough information. The insights presented in this study have the potential to advance the management and enrichment of comprehensive bibliographic databases.

Acknowledgments and Disclosure of Funding

We thank Florian Reitz from DBLP for valuable feedback and input to the research, most importantly the scenario and critical reflection of the impact of the work for bibliographic metadata provider. The authors acknowledge support by the state of Baden- Württemberg through bwHPC. This research is co-funded by the SmartER project (No. 515537520) of the DFG, German Research Foundation.

References

Zahra Abbasiantaeb and Saeedeh Momtazi. Text-based question answering from information retrieval and deep neural network perspectives: A survey. *WIREs Data Mining Knowl. Discov.*, 11(6), 2021. doi: 10.1002/WIDM.1412.

Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David A. Sontag. Large language models are few-shot clinical information extractors. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP* 2022, pages 1998–2022. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.EMNLP-MAIN.130.

Sophia Althammer, Sebastian Hofstätter, Mete Sertkan, Suzan Verberne, and Allan Hanbury. PARM: A paragraph aggregation retrieval model for dense document-to-document retrieval. In *Advances in Information Retrieval - 44th European Conference on IR Re-*

search, ECIR 2022, Proceedings, Part I, volume 13185 of *Lecture Notes in Computer Science*, pages 19–34. Springer, 2022. doi: 10.1007/978-3-030-99736-6_2.

Xavier Amatriain. Prompt design and engineering: Introduction and advanced methods. *CoRR*, abs/2401.14423, 2024. doi: 10.48550/ARXIV.2401.14423.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *CoRR*, abs/2005.14165, 2020. URL `https://arxiv.org/abs/2005.14165`.

Catherine Chen, Zejiang Shen, Dan Klein, Gabriel Stanovsky, Doug Downey, and Kyle Lo. Are layout-infused language models robust to layout distribution shifts? A case study with scientific documents. In *Findings of the Association for Computational Linguistics: ACL* 2023, pages 13345–13360. Association for Computational Linguistics, 2023. doi: 10.18653/V1/2023.FINDINGS-ACL.844.

Changyou Chen, Jianyi Zhang, Yi Xu, Liqun Chen, Jiali Duan, Yiran Chen, Son Tran, Belinda Zeng, and Trishul Chilimbi. Why do we need large batchsizes in contrastive learning? A gradient-bias perspective. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS*, 2022. URL `http://papers.nips.cc/paper_files/paper/2022/hash/db174d373133dcc6bf83bc98e4b681f8-Abstract-Conference.html`.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416, 2022. doi: 10.48550/ARXIV.2210.11416.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423.

Huy Hoang Nhat Do, Muthu Kumar Chandrasekaran, Philip S. Cho, and Min-Yen Kan. Extracting and matching authors and affiliations in scholarly documents. In *13th ACM/IEEE-CS Joint Conference on Digital Libraries, JCDL 2013*, pages 219–228. ACM, 2013. doi: 10.1145/2467696.2467703.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. A survey on RAG meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024*, pages 6491–6501. ACM, 2024. doi: 10.1145/3637528.3671470.

Michael Günther, Jackmin Ong, Isabelle Mohr, Alaeddine Abdessalem, Tanguy Abel, Mohammad Kalim Akram, Susana Guzman, Georgios Mastrapas, Saba Sturua, Bo Wang, Maximilian Werk, Nan Wang, and Han Xiao. Jina embeddings 2: 8192-token general-purpose text embeddings for long documents. *CoRR*, abs/2310.19923, 2023. doi: 10.48550/ARXIV.2310.19923.

Jiafeng Guo, Yixing Fan, Liang Pang, Liu Yang, Qingyao Ai, Hamed Zamani, Chen Wu, W. Bruce Croft, and Xueqi Cheng. A deep look into neural ranking models for information retrieval. *Inf. Process. Manag.*, 57(6):102067, 2020. doi: 10.1016/J.IPM.2019.102067.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. REALM: retrieval-augmented language model pre-training. *CoRR*, abs/2002.08909, 2020. URL `https://arxiv.org/abs/2002.08909`.

Hui Han, C. Lee Giles, Eren Manavoglu, Hongyuan Zha, Zhenyue Zhang, and Edward A. Fox. Automatic document metadata extraction using support vector machines. In *ACM/IEEE 2003 Joint Conference on Digital Libraries (JCDL 2003), 27-31 May 2003, Houston, Texas, USA, Proceedings*, pages 37–48. IEEE Computer Society, 2003. doi: 10.1109/JCDL.2003.1204842.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry P. Heck. Learning deep structured semantic models for web search using clickthrough data. *22nd ACM International Conference on Information and Knowledge Management, CIKM'13*, pages 2333–2338. ACM, 2013. doi: 10.1145/2505515.2505665.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. Layoutlmv3: Pre-training for document AI with unified text and image masking. In *MM '22: The 30th ACM International Conference on Multimedia*, pages 4083–4091. ACM, 2022. doi: 10.1145/3503161.3548112.

Gautier Izacard, Patrick S. H. Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models. *J. Mach. Learn. Res.*, 24: 251:1–251:43, 2023. URL `http://jmlr.org/papers/v24/23-0037.html`.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, 2002. doi: 10.1145/582415.582418.

Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020*, pages 39–48. ACM, 2020. doi: 10.1145/3397271.3401075.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, 2020. URL `https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html`.

Michael Ley. DBLP - some lessons learned. *Proc. VLDB Endow.*, 2(2):1493–1500, 2009. doi: 10.14778/1687553.1687577.

Huajing Li, Isaac G. Councill, Wang-Chien Lee, and C. Lee Giles. Citeseerx: an architecture and web service design for an academic document search engine. In *Proceedings of the 15th international conference on World Wide Web, WWW 2006*, pages 883–884. ACM, 2006. doi: 10.1145/1135777.1135926.

Minghan Li and Éric Gaussier. Intra-document block pre-ranking for bert-based long document information retrieval - abstract. In *Proceedings of the 2nd Joint Conference of the Information Retrieval Communities in Europe (CIRCLE 2022), 2022*, volume 3178 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022. URL `https://ceur-ws.org/Vol-3178/CIRCLE_2022_paper_27.pdf`.

Xi Victoria Lin, Xilun Chen, Mingda Chen, Weijia Shi, Maria Lomeli, Richard James, Pedro Rodriguez, Jacob Kahn, Gergely Szilvasy, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. RA-DIT: retrieval-augmented dual instruction tuning. In *The Twelfth International Conference on Learning Representations, ICLR 2024*. OpenReview.net, 2024. URL `https://openreview.net/forum?id=22OTbutug9`.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Trans. Assoc. Comput. Linguistics*, 12:157–173, 2024. doi: 10.1162/TACL_A_00638.

Patrice Lopez. GROBID: combining automatic bibliographic data recognition and term extraction for scholarship publications. In *Research and Advanced Technology for Digital Libraries, 13th European Conference, ECDL 2009,Proceedings*, volume 5714 of *Lecture Notes in Computer Science*, pages 473–474. Springer, 2009. doi: 10.1007/978-3-642-04346-8_62.

Zhenyan Lu, Xiang Li, Dongqi Cai, Rongjie Yi, Fangming Liu, Xiwen Zhang, Nicholas D. Lane, and Mengwei Xu. Small language models: Survey, measurements, and insights. *CoRR*, abs/2409.15790, 2024. doi: 10.48550/ARXIV.2409.15790.

Yuanhua Lv and ChengXiang Zhai. Lower-bounding term frequency normalization. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM 2011*, pages 7–16. ACM, 2011. doi: 10.1145/2063576.2063584.

Sean MacAvaney, Andrew Yates, Arman Cohan, and Nazli Goharian. CEDR: contextualized embeddings for document ranking. In *Proceedings of the 42nd International ACM SIGIR*

Conference on Research and Development in Information Retrieval, SIGIR 2019, pages 1101–1104. ACM, 2019. doi: 10.1145/3331184.3331317.

Bhaskar Mitra and Nick Craswell. An introduction to neural information retrieval. *Found. Trends Inf. Retr.*, 13(1):1–126, 2018. doi: 10.1561/1500000061.

Ha-Thanh Nguyen, Manh-Kien Phi, Xuan-Bach Ngo, Vu Tran, Le-Minh Nguyen, and Minh-Phuong Tu. Attentive deep neural networks for legal document retrieval. *Artif. Intell. Law*, 32(1):57–86, 2024. doi: 10.1007/S10506-022-09341-8.

OpenAI. GPT-4 technical report. *CoRR*, abs/2303.08774, 2023. doi: 10.48550/arXiv.2303.08774.

Shivam Parmar and Hemil Patel. Prompt engineering for large language model. 2024. doi: 10.13140/RG.2.2.11549.93923.

Vincent Perot, Kai Kang, Florian Luisier, Guolong Su, Xiaoyu Sun, Ramya Sree Boppana, Zilong Wang, Jiaqi Mu, Hao Zhang, and Nan Hua. LMDX: language model-based document information extraction and localization. *CoRR*, abs/2309.10952, 2023. doi: 10.48550/ARXIV.2309.10952.

Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *The Tenth International Conference on Learning Representations, ICLR 2022*. OpenReview.net, 2022. URL `https://openreview.net/forum?id=R8sQPpGCv0`.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. In-context retrieval-augmented language models. *Trans. Assoc. Comput. Linguistics*, 11:1316–1331, 2023. doi: 10.1162/TACL_A_00605.

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. In Donna K. Harman, editor, *Proceedings of The Third Text REtrieval Conference, TREC 1994*, volume 500-225 of *NIST Special Publication*, pages 109–126. National Institute of Standards and Technology (NIST), 1994. URL `http://trec.nist.gov/pubs/trec3/papers/city.ps.gz`.

Tarek Saier, Mayumi Ohta, Takuto Asakura, and Michael Färber. Hyperpie: Hyperparameter information extraction from scientific publications. *CoRR*, abs/2312.10638, 2023. doi: 10.48550/ARXIV.2312.10638.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. Colbertv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022*, pages 3715–3734. Association for Computational Linguistics, 2022. doi: 10.18653/V1/2022.NAACL-MAIN.272.

Ralf Schenkel. Integrating and exploiting public metadata sources in a bibliographic information system. In *Proceedings of the 7th International Workshop on Bibliometric-enhanced Information Retrieval (BIR 2018) co-located with the 40th European Conference*

on Information Retrieval (ECIR 2018), volume 2080 of *CEUR Workshop Proceedings*, pages 16–21. CEUR-WS.org, 2018. URL https://ceur-ws.org/Vol-2080/paper2.pdf.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=RIu5lyNXjT.

Mahsa Shamsabadi, Jennifer D'Souza, and Sören Auer. Large language models for scientific information extraction: An empirical study for virology. *CoRR*, abs/2401.10040, 2024. doi: 10.48550/ARXIV.2401.10040.

Xuedong Tian and Jiameng Wang. Retrieval of scientific documents based on HFS and BERT. *IEEE Access*, 9:8708–8717, 2021. doi: 10.1109/ACCESS.2021.3049391.

Dominika Tkaczyk, Pawel Szostek, Mateusz Fedoryszak, Piotr Jan Dendek, and Lukasz Bolikowski. CERMINE: automatic extraction of structured metadata from scientific literature. *Int. J. Document Anal. Recognit.*, 18(4):317–335, 2015. doi: 10.1007/S10032-015-0249-8.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018. URL http://arxiv.org/abs/1807.03748.

Dongsheng Wang, Natraj Raman, Mathieu Sibue, Zhiqiang Ma, Petr Babkin, Simerjot Kaur, Yulong Pei, Armineh Nourbakhsh, and Xiaomo Liu. Docllm: A layout-aware generative language model for multimodal document understanding. *CoRR*, abs/2401.00908, 2024a. doi: 10.48550/ARXIV.2401.00908.

Jiajia Wang, Jimmy Xiangji Huang, Xinhui Tu, Junmei Wang, Angela Jennifer Huang, Md. Tahmid Rahman Laskar, and Amran Bhuiyan. Utilizing BERT for information retrieval: Survey, applications, resources, and challenges. *ACM Comput. Surv.*, 56(7):185:1–185:33, 2024b. doi: 10.1145/3648471.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, and Enhong Chen. Large language models for generative information extraction: A survey. *CoRR*, abs/2312.17617, 2023. doi: 10.48550/ARXIV.2312.17617.

Ye Zhang, Md. Mustafizur Rahman, Alex Braylan, Brandon Dang, Heng-Lu Chang, Henna Kim, Quinten McNamara, Aaron Angert, Edward Banner, Vivek Khetan, Tyler McDonnell, An Thanh Nguyen, Dan Xu, Byron C. Wallace, and Matthew Lease. Neural information retrieval: A literature review. *CoRR*, abs/1611.06792, 2016. URL http://arxiv.org/abs/1611.06792.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Zhicheng Dou, and Ji-Rong Wen. Large language models for information retrieval: A survey. *CoRR*, abs/2308.07107, 2023. doi: 10.48550/ARXIV.2308.07107.

Appendix A. Publisher Distribution

Table 13 shows the distribution of publications among the 25 largest publishers on DBLP. The table includes the number of publications for each publisher, the accumulated count, and the accumulated coverage of all publications on DBLP. For our experiments, we selected the six largest publishers, as they collectively cover more than 80% of all publications listed on DBLP.

Count	Acc. Count	Acc. Coverage	DOI Prefix	Publisher
2,058,673	2,058,673	35.86%	10.1109	Institute of Electrical and Electronics Engineers
1,103,631	3,162,304	55.08%	10.1007	Springer-Verlag
661,729	3,824,033	66.61%	10.1016	Elsevier
531,375	4,355,408	75.86%	10.1145	Association for Computing Machinery
169,109	4,524,517	78.81%	10.48550	arXiv
135,468	4,659,985	81.17%	10.3390	MDPI AG
81,140	4,741,125	82.58%	10.1002	Wiley Blackwell (John Wiley & Sons)
64,646	4,805,771	83.71%	10.1080	Informa UK (Taylor & Francis)
53,450	4,859,221	84.64%	10.1093	Oxford University Press
52,646	4,911,867	85.56%	10.3233	IOS Press
43,017	4,954,884	86.31%	10.1137	Society for Industrial and Applied Mathematics
42,654	4,997,538	87.05%	10.1142	World Scientific
39,990	5,037,528	87.75%	10.1504	Inderscience Enterprises Ltd.
39,090	5,076,618	88.43%	10.1155	Hindawi Publishing Corporation
32,696	5,109,314	89.00%	10.23919	Institute of Electrical and Electronics Engineers
31,807	5,141,121	89.55%	10.1186	Springer (Biomed Central Ltd.)
30,875	5,171,996	90.09%	10.4018	IGI Global
26,018	5,198,014	90.54%	10.18653	Association for Computational Linguistics
25,522	5,223,536	90.99%	10.1117	SPIE - International Society for Optical Engineering
24,623	5,248,159	91.41%	10.21437	International Speech Communication Association
24,606	5,272,765	91.84%	10.1108	Emerald (MCB UP)
23,946	5,296,711	92.26%	10.5220	Scitepress
23,313	5,320,024	92.67%	10.1287	Institute for Operations Research and the Management Sciences
23,017	5,343,041	93.07%	10.1177	Sage Publications
22,163	5,365,204	93.45%	10.1111	Wiley Blackwell (Blackwell Publishing)

Table 13: Publisher distribution on DBLP for the 25 biggest publisher. Showing the number of publications listed on DBLP, the accumulated count and the accumulated coverage of all publications on DBLP.

Appendix B. Per-Publisher Ranking Baseline Comparison

Table 14 presents a detailed breakdown of retrieval performance for each baseline method across the six publishers. The fine-tuned Jina-v2 retriever outperforms all baselines for every publisher. BM25 and BM25+ perform comparably on most publishers but both struggle particularly with Springer and ACM. The zero-shot Jina-v2 embeddings show the weakest performance across all publishers, confirming that domain-specific fine-tuning is essential for our task.

Publisher	Jina-v2 (zero-shot)			BM25			BM25+			Jina-v2 (ours)		
	MRR	MAP	nDCG	MRR	MAP	nDCG	MRR	MAP	nDCG	MRR	MAP	nDCG
IEEE	0.082	0.085	0.338	0.328	0.187	0.507	0.328	0.197	0.513	1.000	1.000	1.000
Springer	0.024	0.225	0.168	0.134	0.340	0.276	0.178	0.387	0.335	0.800	0.998	0.800
Elsevier	0.027	0.032	0.234	0.465	0.218	0.438	0.486	0.232	0.460	1.000	0.970	0.985
ACM	0.055	0.110	0.348	0.201	0.119	0.369	0.207	0.142	0.393	1.000	0.999	1.000
arXiv	0.062	0.085	0.347	0.333	0.253	0.513	0.329	0.248	0.508	1.000	1.000	1.000
MDPI	0.018	0.028	0.246	0.864	0.348	0.604	0.864	0.399	0.650	1.000	0.954	0.982
Average	0.045	0.094	0.280	0.387	0.244	0.451	0.399	0.268	0.477	0.967	0.987	0.961

Table 14: Per-publisher comparison of document ranking performance across retrieval methods. Our fine-tuned Jina-v2 retriever outperforms all baselines for every publisher. Values for the fine-tuned model are from Table 1.

Appendix C. Prompt Design

The design of prompts plays a crucial role in determining the performance of language models. It is widely recognized that even minor modifications in prompt design can have a substantial effect on the behavior of the model (Sclar et al., 2024). Prompt effectiveness can also vary across different models. Therefore, we designed a prompt specifically tailored to our task and model, following established best practices (Amatriain, 2024; Parmar and Patel, 2024). Our prompt utilizes an XML-style structure as proposed by Perot et al. (2023) to clearly define the task and desired output, providing a structured and machine-readable format. We also included specific metadata fields as context amplification, following the recommendations of Parmar and Patel (2024). Through iterative experiments, we fine-tuned the prompt (presented in Listing 1) to achieve optimal performance.

Furthermore, we discovered that the order of prompt components played a crucial role in model performance. In our experiments, we found that presenting the task description and output format first as a system prompt, followed by the context as a user prompt yielded the best results. This arrangement allowed the model to maintain a clear focus on the task.

Appendix D. Data Labeling Tool

The process of labeling data for this task required a substantial amount of manual labor. To make the process more efficient, we created a specialized utility tool. Our tool employs a two-stage approach: In the first stage, we emulate a browser landing page and prompt the user for metadata input (as illustrated in Figure 5). The second stage involves navigating through all linked websites within the browser emulation and asking the labeler to indicate the relevance of each website to the publication. The labeler assigns a binary relevance score, with '1' indicating relevance and '0' indicating irrelevance (as shown in Figure 6). To expedite the labeling process, we utilized regular expression blacklists and whitelists to automatically label common websites, such as cookie policy pages, as irrelevant. Figure 7 provides an overview of the entire setup of our labeling tool, showcasing its user-friendly design and functionality.

```
<PROMPT>
<TASK_INSTRUCTIONS>
<TASK_DESCRIPTION>
The task is to extract bibliographic metadata from the provided context
    while utilizing the layout information. The metadata should be
    structured as follows:
</TASK_DESCRIPTION>

<METADATA_FIELDS>
- title: The title of the publication
- authors: A list of authors with their according names and affiliations
- publication_year: Year of publication
</METADATA_FIELDS>

<OUTPUT_FORMAT>
Provide the extracted metadata in the following JSON format:
{
  "title": "Title of the paper",
  "authors": [
    {
      "name": "Name of the first author",
      "affiliations": [
        "First affiliation of author 1"
      ]
    },
    {
      "name": "Name of the second author",
      "affiliations": [
        "First affiliation of author 2",
        "Second affiliation of author 2"
      ]
    }
  ],
  "publication_date": "YYYY",
}
</OUTPUT_FORMAT>
</TASK_INSTRUCTIONS>
</PROMPT>
```

Listing 1: The information extraction prompt we designed for our task.

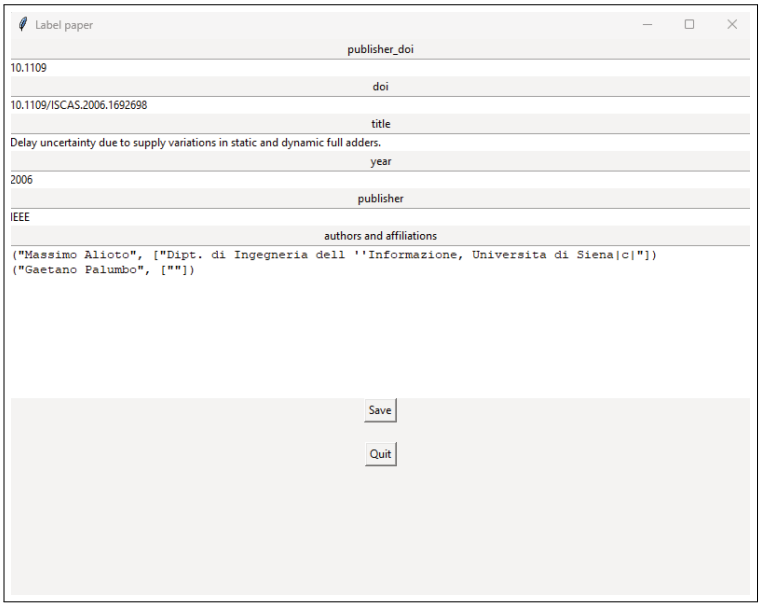


Figure 5: The GUI of our metadata labeling tool

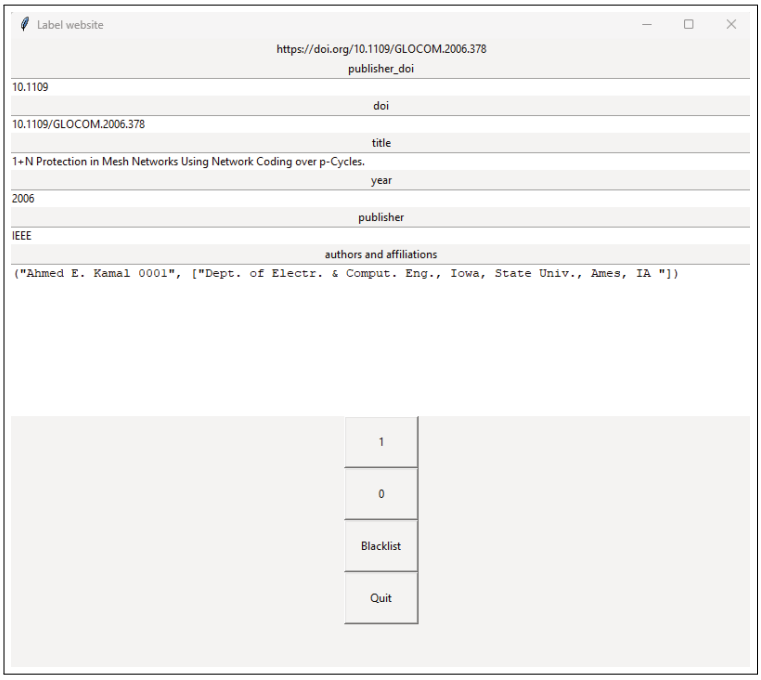


Figure 6: The GUI of our relevancy labeling tool

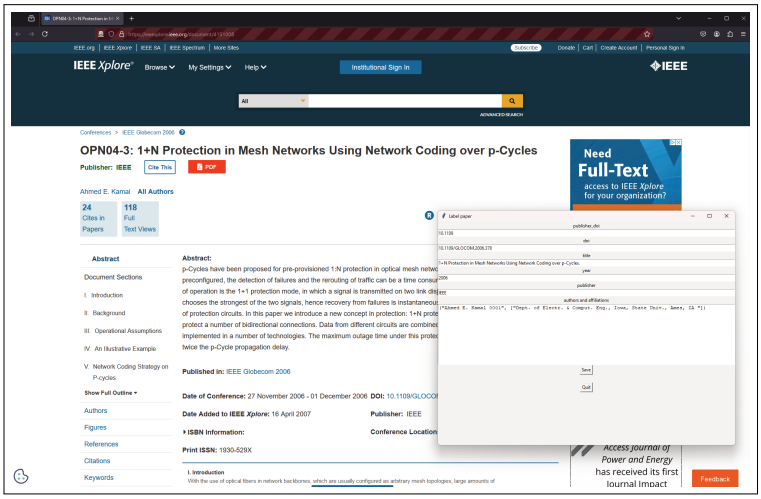


Figure 7: An emulated Firefox browser with the labeling tool

k	Recall@k	Precision@k	F1@k
1	0.34	0.97	0.51
2	0.55	0.89	0.68
3	0.69	0.82	0.75
4	0.79	0.75	0.77
5	0.87	0.69	0.77
6	0.90	0.62	0.73
7	0.93	0.56	0.70
8	0.94	0.50	0.65
9	0.94	0.46	0.61
10	0.95	0.42	0.58
11	0.95	0.38	0.55
12	0.96	0.36	0.52
13	0.96	0.33	0.49
14	0.96	0.31	0.47
15	0.96	0.29	0.45
16	0.96	0.27	0.43
17	0.96	0.26	0.41
18	0.96	0.24	0.39
19	0.96	0.23	0.37
20	0.97	0.22	0.36

Table 15: Ranking performance evaluation of CRAWLDoc at different cut-off values (k).

Appendix E. Additional Analysis of Ranking Performance

This appendix provides supplementary data and analysis to support the findings presented in the main body of the paper. We offer two tables that expand on the results discussed in the primary text. Table 15 provides a detailed evaluation of the model's ranking performance across various cut-off values (k). This Table complements the analysis presented in Figure 4 in the main body of the paper. This table serves as an additional resource to better understand the trends discussed in Section 5.1.

Table 16 presents a more detailed version of Table 4 from the main paper, which includes the standard deviation.

	Title			Author Names			Affiliations			Publication Year		
Publisher	**BLEU**	**P**	**R**	**BLEU**	**P**	**R**	**BLEU**	**P**	**R**	**BLEU**	**P**	**R**
IEEE	1.000	1.000	1.000	0.784	1.000	0.884	0.821	1.000	0.863	1.000	1.000	1.000
	(0.00)	(0.00)	(0.00)	(0.29)	(0.00)	(0.19)	(0.34)	(0.00)	(0.27)	(0.00)	(0.00)	(0.00)
Springer	0.925	1.000	0.946	0.875	1.000	0.899	0.955	1.000	0.971	0.900	1.000	0.960
	(0.23)	(0.00)	(0.16)	(0.32)	(0.00)	(0.26)	(0.10)	(0.00)	(0.09)	(0.30)	(0.00)	(0.12)
Elsevier	0.989	1.000	0.992	0.926	1.000	0.975	0.932	0.963	0.938	1.000	1.000	1.000
	(0.03)	(0.00)	(0.02)	(0.22)	(0.00)	(0.06)	(0.24)	(0.19)	(0.22)	(0.00)	(0.00)	(0.00)
ACM	0.946	1.000	1.000	0.670	0.838	0.710	0.231	0.649	0.381	1.000	1.000	1.000
	(0.16)	(0.00)	(0.00)	(0.45)	(0.37)	(0.43)	(0.32)	(0.48)	(0.34)	(0.00)	(0.00)	(0.00)
arXiv	0.909	1.000	1.000	0.961	1.000	0.977	0.940	1.000	0.987	0.800	1.000	0.925
	(0.17)	(0.00)	(0.00)	(0.14)	(0.00)	(0.08)	(0.09)	(0.00)	(0.03)	(0.40)	(0.00)	(0.15)
MDPI	0.914	1.000	0.936	0.972	1.000	0.994	0.932	1.000	0.987	1.000	1.000	1.000
	(0.26)	(0.00)	(0.19)	(0.13)	(0.00)	(0.04)	(0.08)	(0.00)	(0.02)	(0.00)	(0.00)	(0.00)
All	0.947	1.000	0.979	0.876	0.974	0.913	0.810	0.939	0.865	0.950	1.000	0.981
	(0.17)	(0.00)	(0.11)	(0.29)	(0.16)	(0.24)	(0.33)	(0.24)	(0.29)	(0.22)	(0.00)	(0.08)

Table 16: Extraction performance for the different entity types, with standard deviation. 'P' denotes n-gram precision and 'R' is n-gram recall. Metrics are provided per publisher as well as aggregated over all.

Information Retrieval Research 2 (2026) 35-58 Submitted 8/25; Revised: 2/26; Published 3/26

Simple Techniques for Efficient Top-k Batch Query Processing

Zhixuan Li ZHIXUAN.LI@UQ.EDU.AU
School of Electrical Engineering and Computer Science
The University of Queensland
St Lucia, Queensland, Australia

Joel Mackenzie JOEL.MACKENZIE@UQ.EDU.AU
School of Electrical Engineering and Computer Science
The University of Queensland
St Lucia, Queensland, Australia

Editor: Ismail Sengor Altingovde

Abstract

When faced with a large number of related tasks, like clearing unread emails or tackling a mountain of laundry, it's often most efficient to handle them together as a single *batch*. The simple observation is that grouping similar tasks together can ultimately streamline the entire process. In the context of Information Retrieval, batch processing has been applied to the problem of *conjunctive querying* by re-using partially computed results to avoid redundant list intersections or disk reads, reducing the total computational effort required to answer the given query batch. However, there has yet to be any work on exploiting batch query processing in the context of *disjunctive querying* semantics, which are often implemented by highly optimized top-k dynamic pruning algorithms. In this work, we explore two simple yet efficient approaches to reduce the computational cost of top-k querying in the batch processing regime. Our experimentation on two collections, and two unique query batches, demonstrates end-to-end cost reductions of up to 44% over standard query processing algorithms, and lays the foundation for future work towards more sophisticated top-k batch querying techniques.

Keywords: Batch Processing, Caching, Dynamic Pruning, Experimentation

1 Introduction

Efficiently processing queries is a long-standing problem in the field of Information Retrieval (IR), with work dating back several decades to the 1970s and 1980s (Thiel and Heaps, 1972; Buckley and Lewit, 1985). The motivation is simple; more efficient querying generally means that more queries can be processed within a reasonable amount of time – on a fixed hardware budget – enabling better scalability or reductions in end-to-end electricity costs and carbon emissions. An alternative view is that more efficient querying equates to a better user experience (Schurman and Brutlag, 2009), as more expensive and accurate models can be used under that same fixed resource budget while maintaining a reasonable response latency (Bai et al., 2017).

Although there has been a significant amount of work on improving the efficiency of query processing, most of this work has focused on reducing latency or increasing through-

 WWW: https://irrj.org DOI: 10.54195/irrj.23893

put, two fundamentally important aspects for delivering a high-quality user experience (Tonellotto et al., 2018). However, there has been little focus on efficiency from the perspective of total operational cost, or the associated electrical costs (Kayaaslan et al., 2011; Catena et al., 2016; Blanco et al., 2016) or downstream emissions (Scells et al., 2022; Chowdhury, 2012; Zuccon et al., 2023).

One exception is a thread of work on *batch query processing*, first explored by Ding et al. (2011), focusing on the key observation that there may be an entire class of queries that *do not* require immediate responses. This includes queries issued for refreshing caches, mining data, conducting internal testing, for external clients via API calls, or perhaps now to generate training data. Ding et al. investigated how these non-interactive queries might be processed holistically, as a large batch, to reduce the total computational cost of conjunctive query processing for both disk- and memory-resident indexes.

1.1 Contribution

In this work, we re-investigate the notion of batch query processing in IR. Unlike previous work, which has focused exclusively on *conjunctive querying* (Ding et al., 2011; Mackenzie and Moffat, 2023) – where *all* query terms must be present in order to return a document – we are the first to explore how batch query processing can be used in the context of ranked *disjunctive* querying, one form of top-k retrieval.

We propose two simple methods for improving the efficiency of disjunctive ranked retrieval algorithms in the batch processing regime. Both methods are based on the careful re-use of top-k score thresholds to accelerate future queries; one is a novel *static* method, which tries to find and compute promising thresholds *before* processing the batch; the other is a *dynamic* threshold caching approach which stores thresholds opportunistically for use in later queries (Yafay and Altingovde, 2019).

Our results demonstrate that batch querying costs can be reduced for disjunctive querying, even under highly optimized retrieval algorithms. We also demonstrate that these improvements translate to parallel processing contexts, allowing the time to process the entire batch to be reduced while retaining overall cost savings.

2 Background

Modern commercial search engines serve billions of queries each day over enormous volumes of data, incurring similarly enormous operating costs. Given this extreme scale, they are typically implemented as *multi-stage cascades*, where increasingly sophisticated models are applied to successively smaller subsets of documents to allow fine control over the tension between efficiency and effectiveness (Wang et al., 2010; Gallagher et al., 2019). The earliest phase of this process is known as candidate generation, where the goal is to return a large set of candidate documents for re-ranking, and is typically achieved via simple bag-of-words ranking models. Given a query q containing n unique terms, document d is scored as a sum of weights over those terms:

$$\mathcal{S}_{d,q} = \sum_{i=1}^{n} \mathcal{C}(t_i, d). \tag{1}$$

In this formulation, $\mathcal{C}(t, d)$ represents the contribution (or weight) of term t in document d, and captures any additive scoring function like BM25 (Robertson and Zaragoza, 2009), or modern learned sparse retrieval methods (Yates et al., 2024). It is worth noting that while other scoring formulations exist – perhaps including a static per-document quality prior such as pagerank or spam score (Page et al., 1999; Cormack et al., 2011) – we consider these as orthogonal to our study (Shan et al., 2012; Petri et al., 2013).

Typically, the top-k highest ranking documents are returned for further processing. In some circumstances, this formulation may be used to return documents directly to the end user, and in these cases, a small number of results would be retrieved (such as $k = 10$). For candidate generation, k is more likely to be in the hundreds or thousands. We denote the kth highest similarity score achieved when processing all documents in the collection against q as Θ_k. Note that this formulation is inherently *disjunctive* – there is no guarantee that a document will contain all n query terms, and documents can still rank highly even if they only contain a subset of query terms.

In this work, we assume an operating context that emulates a single processing server responsible for candidate generation in a large-scale search engine infrastructure.

2.1 Query Processing

While many index arrangements are possible (Crane et al., 2017; Fontoura et al., 2011), we use the common *document-ordered* index which facilitates fast *document-at-a-time* retrieval. In particular, we assume an inverted index stores a *postings list* for each term in the corpus vocabulary, and these postings lists contain monotonically increasing document identifiers and corresponding payload information. Those payloads can be assumed to store (quantized) $\mathcal{C}(d, t)$ values, or statistics such as term frequencies that allow $\mathcal{C}$ to be readily computed. Postings may be stored on-disk, in-memory, or in some combination thereof (Fontoura et al., 2011), depending on operational requirements and specifications of the hardware.

Given a query, document-at-a-time processing algorithms iterate across the corresponding postings lists simultaneously, allowing the full score of each document d to be computed before continuing to the next candidate document. *Dynamic pruning* algorithms like MaxScore (Turtle and Flood, 1995), WAND (Broder et al., 2003), or Block-Max WAND (Ding and Suel, 2011) facilitate rapid traversal of inverted indexes, often at a fraction of the cost of exhaustive algorithms (Mallia et al., 2019b; Mackenzie and Moffat, 2020). During processing, dynamic pruning algorithms track the evolving top-k highest scoring documents, using the lowest score of those k documents as a threshold, denoted θ. Any document with a score estimated to be below θ will be bypassed, accelerating retrieval without sacrificing effectiveness — so long as these estimates are *overestimates* of the true document score. In other words, the decision to score a document is made using a fast estimation of that document's true score, so an overestimation will ensure we do not skip scoring any candidate document that may actually form part of the top-k results list.

Document score estimations are typically obtained by storing the upper-bound value of $\mathcal{C}$ for each postings list in the index, and examining combinations of these scores to determine which document to score next. Figure 1 shows an example of this process; different combinations of term upper-bounds, in conjunction with the heap threshold θ,

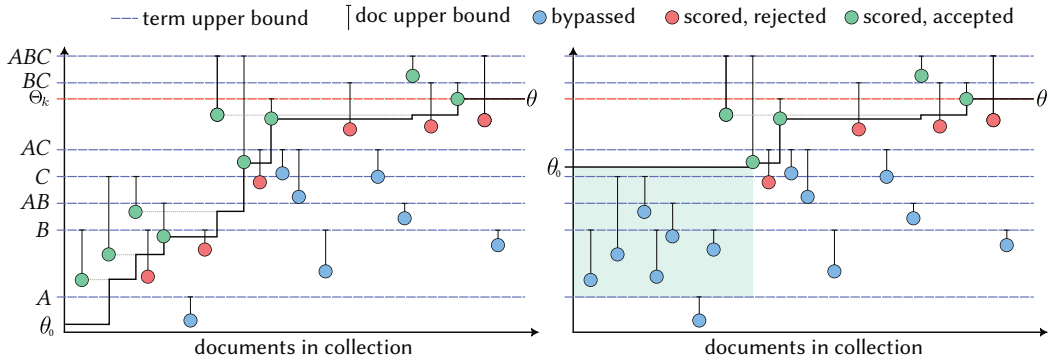


Figure 1: An example of how a dynamic pruning algorithm would find the top $k = 2$ documents on a toy document collection for a three term query (terms A, B, and C). In the left pane, the heap threshold θ grows naturally from zero; in the right pane, the initial threshold θ_0 is *primed* via some non-zero estimate, allowing additional documents to be bypassed (shown with a shaded background). The best possible setting of θ without sacrificing retrieval quality is the terminal value, shown as Θ_k. Figure adapted from Petri et al. (2019).

invoke different levels of skipping; the closer the value of θ to its terminal value, Θ_k, the faster retrieval will be.

This is the key direction of our investigation, and in the following paragraphs, we describe a number of approaches that can use specific properties of the query batch to estimate the *initial* value of θ, denoted θ_0, to accelerate query processing. We refer the interested reader to the work of Petri et al. (2013) and the survey of Tonellotto et al. (2018) for a more comprehensive description of dynamic pruning algorithms.

2.2 Batch Query Processing

Given a batch of queries $\mathcal{B}$, the goal is to return answers to all individual queries $q \in \mathcal{B}$ at a minimal total cost (where this cost is defined according to some metrics of interest such as total data read from disk, or total CPU cycles spent). Ding et al. (2011) were the first to explore batch processing for information retrieval, demonstrating considerable gains for ranked conjunctive queries on both disk and memory-resident inverted indexes (Zobel and Moffat, 2006). These gains were primarily due to the observation that, in batch processing, all queries are known ahead of time. This enables optimizations that would not otherwise be possible over query streams where future queries are not known, including reordering the sequence of queries, optimizing cache behavior, and re-using list intersections to reduce redundant processing. Recently, Mackenzie and Moffat (2023) re-examined this problem in terms of unranked conjunctions, demonstrating further improvements are possible by modeling the cost-vs-benefit of caching specific *pairs* of terms.

Despite the utility of these approaches – shown to achieve up to 2× speedups over standard query-by-query processing – top-k retrieval algorithms have not yet been explored in the batch processing regime. However, since top-k retrieval algorithms typically employ

disjunctive term matching semantics, previous batch optimization techniques based on list intersections are not viable. One exception is the work of Choudhury et al. (2018), which focuses on top-k *spatial-textual* queries for disk-resident indexes. However, their enhancements are tailored towards geospatial indexes, not classic text retrieval problems.

3 Fast Disjunctive Batch Processing

In this section, we outline a series of approaches that can be applied to the problem of disjunctive batch query processing, including simple baselines and novel offline and online processing mechanisms.

3.1 Query-at-a-Time Retrieval

The most obvious baseline, which we denote Naïve, is to simply process each query $q \in \mathcal{B}$ one-by-one with a highly optimized top-k processing algorithm. This has the same characteristics as a typical online retrieval system, and makes no use of the properties of $\mathcal{B}$ to improve efficiency. For each unique query, the initial value of the heap threshold, θ_0, will be initialized as 0, and will grow organically as the query is processed. It is worth noting that, although this algorithm does not make use of $\mathcal{B}$, it is embarrassingly parallel in nature, allowing it to easily scale up with the number of available CPUs if a reduced end-to-end batch-level *latency* is desired.

3.2 Clairvoyant Thresholding

On the other end of the spectrum, we also measure the cost of a Clairvoyant algorithm which "knows" the terminal — and thus optimal — threshold. While this algorithm is not attainable in practice, it represents the best possible speedup of the processing algorithm under threshold estimation; this is a useful yardstick since our primary focus in this work is on threshold-based techniques.

The Clairvoyant algorithm proceeds as follows. Before processing a given query, θ_0 is initialized to the query's corresponding Θ_k value, allowing the processing algorithm to bypass the maximal number of documents (see Figure 1). That is, each Θ_k is pre-computed and is assumed to be freely available for use by the Clairvoyant algorithm.

3.3 Deterministic Term-Based Thresholding

Although the clairvoyant baseline is not attainable in practice, other deterministic estimators are. The most simple of these augments the inverted index with the kth highest impact score for *every* term in the vocabulary for a set of common values of k (such as 10, 100, and 1,000) (Petri et al., 2019; Kane and Tompa, 2018; de Carvalho et al., 2015). Then, θ_0 can be set by taking the *maximum* of these kth highest pre-computed impacts across the terms in the query. This estimator is guaranteed to be safe, as taking the kth highest score for each term assures at least k greater-or-equal scoring documents will be retrieved. This mechanism is referred to as Q_k in our experiments. The key limitation of this approach is that it relies on the value of k being pre-determined, and may not be suitable for situations where k can vary widely.

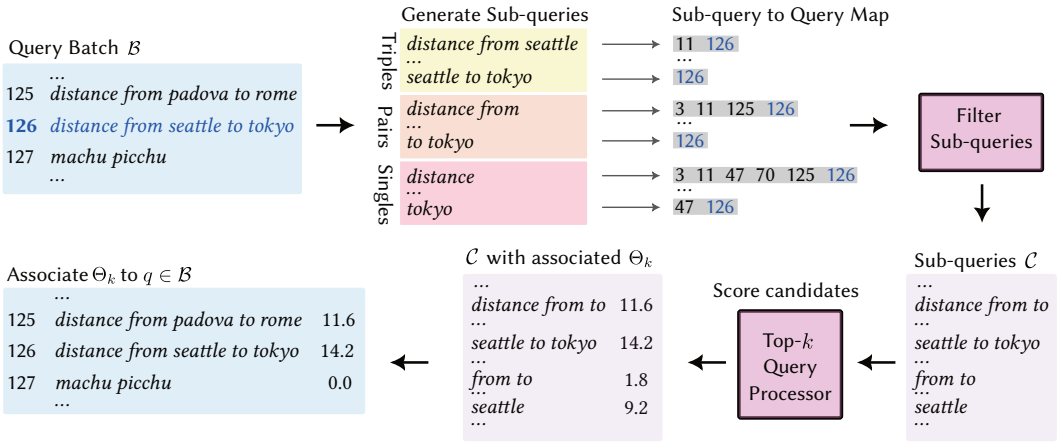


Figure 2: High-level sketch of the *static caching* algorithm with a focus on the query *"distance from seattle to tokyo."* Firstly, all 1-, 2-, and 3-term sub-queries are generated across the entire batch. These are then filtered such that only common sub-queries are considered; these sub-queries are then scored to find their optimal threshold values, and these thresholds are associated back to their parent queries to accelerate processing. In this example, the threshold 14.2 will be used, which was derived from the sub-query *"seattle to tokyo."*

3.4 Static Caching

Since the entire query batch is known ahead of processing, it should be possible to pre-determine a set of thresholds that can be used to expedite the processing of the queries within the batch itself. This approach, denoted *static caching* (SC), aims to associate each query with a best-endeavor threshold based on commonly occurring sub-queries across the entirety of $\mathcal{B}$. The idea is that we can selectively pre-process promising sub-queries that are common to many queries within the batch, with the aim of providing accurate *initial* thresholds to many queries without significant pre-processing costs. The SC algorithm consists of following main conceptual stages, and is outlined in Algorithm 1:

- **Generate**: We first generate all 1-, 2-, and 3-term sub-queries for all $q \in \mathcal{B}$. While computing these sub-queries, we maintain a mapping from sub-queries back to their parent queries (Algorithm 1, lines 1–4). While we could extract longer sub-queries, prior work has found that longer sub-queries are not typically useful as they do not commonly occur (Yafay and Altingovde, 2019).
- **Filter**: Secondly, we ignore any sub-query that occurs in fewer than F queries (Algorithm 1, line 9). The intuition here is that we only want to *pay* for processing a sub-query if it has a good chance of paying itself off later. We denote the remaining candidate query set as $\mathcal{C}$.
- **Score**: Each sub-query in $\mathcal{C}$ is then processed via the top-k retrieval algorithm to find its final threshold value Θ_k (Algorithm 1, line 11). Note that this query runs from a "cold start" – its threshold is initialized to zero.

Algorithm 1 Static Caching Algorithm
Input: An array of all $q \in \mathcal{B}$; a filtering factor F.
Output: The results for each query in $\mathcal{B}$.

```
1:  subqueries ← {}                          ▷ A set of sets mapping each subquery to its associated queries
2:  for q ∈ B do
3:      for all subquery ∈ q where |subquery| = {1, 2, 3} do
4:          append q.id to subqueries[subquery]
5:  thresholds ← {}                                  ▷ Value is initialized to 0 if not found
6:  for subquery in subqueries do                    ▷ Iterate the keys of the subqueries structure
7:      qlist ← subqueries[subquery]                 ▷ Get list of associated query identifiers
8:      C ← ∅
9:      if |qlist| ≥ F then
10:         Add subquery to C
11:         Θ_k ← process_query(subquery, 0)         ▷ Get heap threshold
12:         for qid ∈ qlist do                       ▷ Loop over identifiers of all queries with this subquery
13:             thresholds[qid] ← max(Θ_k, thresholds[qid])
14: for q in B \ C do                                ▷ Process remaining queries
15:     result ← process_query(q, thresholds[q.id])
```

- **Associate**: For each query in the batch that is associated with the current sub-query, we update its threshold if the newly found threshold is larger than the one currently associated (Algorithm 1 lines 12–13).
- **Process**: Finally, we process the entire query batch $\mathcal{B}$ using the thresholds obtained previously to "jump start" query processing (Algorithm 1, lines 14–15).

Figure 2 shows a high-level illustration of the conceptual steps taken by the SC algorithm.

Offline-vs-Online Compute With SC, there is a trade-off between the cost of pre-computing thresholds and the subsequent gains achieved during the final batch processing. If F is too high, the size of the candidate set $\mathcal{C}$ will be small, meaning that pre-processing will be fast but with perhaps little or no gain during batch processing. Conversely, if F is too low, the pre-processing cost of SC may outweigh any benefits arising in the batch processing stage. In our experiments, we deploy SC with three values of F (the number of queries that must contain a sub-query to be retained), namely $F = \{40, 80, 160\}$,[1] meaning that only commonly occurring sub-queries will be utilized; exploring alternative settings is left for future work. It is also worth noting that SC can be thought of as a *pre-processing* step to executing the batch, and may be used in conjunction with further optimizations.

Variant: All Singles We also experimented with a variant of the SC algorithm that pre-computes and caches *all single-term sub-queries* irrespective of their frequency. This is effectively the same as Algorithm 1 except that F is set to zero for single term sub-queries. The intuition is that these singles are likely to be cheap to compute, and appear

1. Selected from preliminary experiments on `MSMARCO-v1` with $F = \{10, 20, 40, 80, 160\}$ – we found that both $F = 10$ and $F = 20$ incurred pre-processing times that heavily outweighed any benefit gained during query processing. We do not experiment with these settings further.

more frequently than term pairs or triples across the batch. However, we found that this all-singles variant was always within about 1-2% of the vanilla SC approach, so we do not report it further.

Parallel Implementation Converting Algorithm 1 to a parallel algorithm is relatively trivial. The most expensive part of the static caching approach is the *scoring* phase, where the thresholds for the selected sub-queries are generated; this can be made parallel by simply rearranging the main `for` loop on line 6 to ensure there are no data races. In particular, we explicitly separate the *score* step from the *association* step on line 12–13 so there are no data races on the *thresholds* hash table, allowing the scoring to be done in a parallel `for` loop. Similarly, the final batch processing operation on line 14 can be converted into a parallel `for` loop.

3.5 Dynamic Caching

One of the key stages in the static caching algorithm involves *scoring* each candidate sub-query to gather its final threshold. Intuitively, processing the candidate sub-queries is no different to processing the query batch $\mathcal{B}$ itself; a query is handed to a top-k retrieval algorithm, and the top-k documents are returned along with their scores. A simple idea follows: Can we re-use the thresholds derived from processing the queries in $\mathcal{B}$ without requiring sub-queries to be computed at all? That is, can we arrange the query batch to maximize the likelihood of re-using the thresholds from previously processed queries?

Previously, Yafay and Altingovde (2019) proposed a family of rank-safe *threshold caching* approaches which store $\langle q', \Theta_k \rangle$ pairs in a hash table during query processing. In this work, we denote this family of methods (DC) to mean *dynamic caching* (as compared to the *static caching* method introduced in Section 3.4). Given a new query q, the hash table is probed for sub-queries of q until a suitable threshold is found, allowing the initial setting of θ to be safely estimated. They proposed three distinct heuristics, outlined as follows:

- DC1: Given a query q, all sub-queries of length 3 are probed; if multiple are found, the one with the maximum Θ_k is used. If no length 3 sub-query is found, we repeat the process for 2-term sub-queries; and then again for 1-term sub-queries. If no sub-query is found, θ_0 will be initialized to 0. Algorithm 2 outlines the operation of the DC1 variant for batch processing.

- DC2: Similar to DC1, all sub-queries of length 3, 2, and 1 are probed. However, instead of backing out once a non-zero threshold is found, this heuristic exhaustively enumerates *all* sub-queries. The intuition here is that, for example, some 1-term sub-queries may still have larger values of Θ_k than their 2-term sub-queries, and so enumerating all possible sub-queries will maximize the value of θ_0; see AB vs C for a concrete example of this in Figure 1. Algorithm 2 can be modified to the DC2 variant by simply omitting lines 8 and 9.

- DC3: Instead of examining 3-, 2-, and 1-term sub-queries, only sub-queries of length $|q|-1$ are probed; if found, these are expected to yield high thresholds, but they are less likely to appear in the map. However, there will also be fewer map look-ups, as the number of candidate sub-queries is linear in $|q|$. Algorithm 2 can be modified to the DC2 variant by modifying line 5 to examine only queries with $l = |q| - 1$.

Algorithm 2 Dynamic Caching Algorithm (Variant DC1)
Input: An array of all $q \in \mathcal{B}$.
Output: The results for each query in $\mathcal{B}$.

```
1:  B ← sort(B)                          ▷ On query length, ascending, and then lexicographically
2:  thresholds ← {}                      ▷ Maps queries to their final threshold
3:  for q ∈ B do
4:      θ_0 ← 0
5:      for l ∈ {3, 2, 1} do             ▷ Enumerate sub-queries of length l
6:          for all q' ∈ q where |q'| ≡ l do
7:              θ_0 ← max(θ_0, thresholds[q'])
8:          if θ_0 > 0 then              ▷ If a non-zero threshold was found, stop searching
9:              break
10:     ⟨result, Θ_k⟩ ← process_query(q, θ_0)
11:     thresholds[q] ← Θ_k
12:     output result
```

Adapting this technique to batch processing, we maintain a dynamic threshold cache, where each $\langle q, \Theta_k \rangle$ pair is added to this cache immediately after processing q. Observe, however, that the *order* in which queries are processed will have a large influence on the effectiveness of this approach, as a given query q can only benefit from the threshold cache if a previously executed query is a sub-query of q. As such, an important pre-processing step is required – we can optimize the cache behavior of the DC algorithms by simply sorting $\mathcal{B}$ in *ascending order of query length*, and then lexicographically *within* each length group, thereby maximizing the threshold cache hit rate. This simple modification to the original DC approach allows it to be tailored to the batch processing context, exploiting the fact that the full set of queries is known ahead of time.

Parallel Implementation Converting the DC algorithms into parallel versions is slightly more complex than for their static counterpart. Observe that in the main processing loop, the *thresholds* hash table is read for a set of sub-queries from each query (Algorithm 2, line 7). However, that same structure is also written to once each query has been processed (Algorithm 2, line 11), causing a data race when multiple readers/writers are present. While synchronization could be achieved via locking, this would significantly impact the efficiency of the algorithm.

Instead, we proceed by treating the querying process as a series of "rounds" – each made up of queries sharing the same length – and applying an explicit synchronization step between each round. In short, the parallel algorithm proceeds as follows. All single term queries are processed in parallel, and the resulting thresholds are maintained in a temporary array. Then, before processing all two term queries, these thresholds are propagated into the global *thresholds* cache (hash table). This process repeats for two term queries, and again for three term queries, and so on, treating each set of queries with the same length as a discrete, parallelizable operation. The key observation is that for the thresholds cache to be effective, it only needs to contain the thresholds for queries that are *shorter* than those currently being processed. Synchronization is fast, and simply involves inserting the

Table 1: Index statistics for the two document corpora used in our experiments. Note that both collections are *passage* collections, with relatively short documents.

Collection	Documents	Unique Terms	Postings	Size (GiB)
MSMARCO-v1	8,841,823	2,660,824	266,247,718	0.9
MSMARCO-v2	138,364,198	16,579,071	8,629,430,400	22.8

key/value pairs from the temporary array into the global *thresholds* table for the next round of processing.

4 Experimental Setup

Before discussing our results, we first outline our experimental setup and methodology.

4.1 Hardware

All experiments are conducted entirely in-memory on a Linux server with a 3.6GHz AMD Threadripper Pro 5975WX and 512GiB of RAM. Unless otherwise specified, experiments use a single processing thread on an otherwise idle machine, allowing end-to-end latency to be used as a proxy for total computing effort.

4.2 Document Collections

Following the recent batch processing study from Mackenzie and Moffat (2023), we use the MSMARCO-v1 and MSMARCO-v2 (Bajaj et al., 2018) passage collections, containing 8.8 and 138.4 million passages, respectively (see Table 1).

4.3 Query Batches

Since we expect batch processing to be useful for large query sets, we make use of the ORCAS (Craswell et al., 2020) query log as our primary batch. Following the same methodology of Mackenzie and Moffat (2023), the entire set of 10 million ORCAS queries were normalized (stopped, stemmed, case-folded) with the default English Lucene tokenizer. Then, all within-query duplicate terms were removed, and only queries with no out-of-vocabulary terms on both indexes were retained. This resulted in a batch containing 6,761,892 unique queries with 3.2 terms per query on average. We also make use of the training queries from the MSMARCO Web Search (MSM-WS) collection (Chen et al., 2024); more detail is provided in Section 5.7. Figure 3 plots the distribution of query lengths across each of these logs. While the distributions are similar, the ORCAS batch has a higher proportion of two-term queries, and a lower proportion of four-term queries.

4.4 Software and Settings

Our experiments were implemented within the efficient PISA system (Mallia et al., 2019a) and compiled using gcc 11.4.0 with -O3 optimization. We used the software from Yafay and Altingovde (2019) to re-implement their DC mechanisms. Each collection was indexed

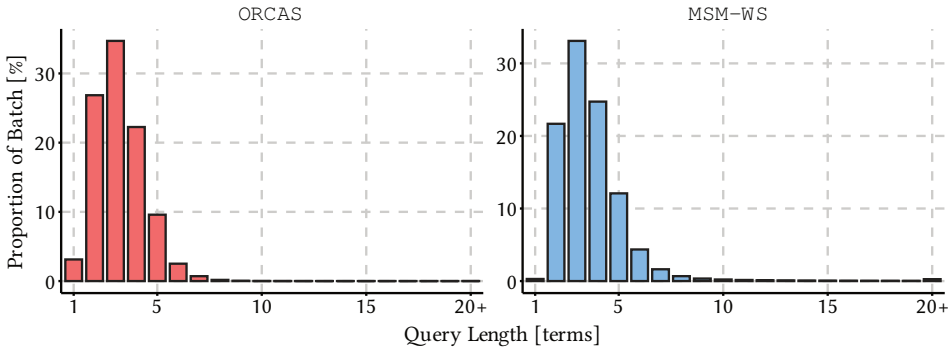


Figure 3: The distribution of query lengths across both the ORCAS (left, 6.76 million queries) and the MSMARCO Web Search (right, 3.83 million queries) query batches.

using Anserini (Yang et al., 2018) and converted to PISA with the CIFF tool (Lin et al., 2020). PISA indexes were reordered with bipartite graph partitioning (Dhulipala et al., 2016; Mackenzie et al., 2022), quantized to 8 bits (Crane et al., 2013) with PISA's BM25 variant (Robertson and Zaragoza, 2009; Kamphuis et al., 2020), and compressed with SIMD-BP128 (Lemire and Boytsov, 2015).

For query processing, we use two representative algorithms: the MaxScore algorithm (Turtle and Flood, 1995); and the highly optimized *Variable* Block-Max WAND (VBMW) algorithm (Mallia et al., 2017), with an average block size of 40 ± 0.5 elements. These settings were based on best practices recommended by Mallia et al. (2019b) and Mackenzie and Moffat (2020).

5 Experiments

This section describes our empirical experimentation, and subsequent analysis, focusing on overall time savings and space overheads, before conducting further analysis on the behavior of the proposed algorithms.

5.1 Time Improvements

Our first experiment applies each batch querying mechanism to process the entire ORCAS batch on both MSMARCO-v1 and MSMARCO-v2. The main metric of interest is the total time, in seconds, to process the batch, with lower values representing a more efficient processing regime. Table 2 and Table 3 present the end-to-end time required to process all 6.76 million queries in the batch for the algorithms we tested, on both small and large settings of k, the number of results to retrieve.

Firstly, we observe that the algorithmic Q_k predictor is a strong baseline, leading to latency reductions of between 4% and 33% over the naïve baseline.

The *static caching* algorithms manage to reduce querying latency further – up to 35%, with larger improvements observed with larger values of k. These improvements represent savings of up to 570 *minutes* worth of CPU time in the best case. Interestingly, there is

Table 2: Total time (including all pre-processing operations) and relative latency change over the Naïve baseline when executing all 6.76 million queries on both MSMARCO-v1 and MSMARCO-v2 indexes, using the MaxScore query processing algorithm.

(a) Setting: top-$k = 10$

Strategy	MSMARCO-v1		MSMARCO-v2	
	Total (sec)	Change (%)	Total (sec)	Change (%)
Naïve	3258	—	25,205	—
Q_k	2851	↓ 12.5	23,354	↓ 7.3
SC $F = 40$	2617	↓ 19.7	25,198	0
SC $F = 80$	2693	↓ 17.3	25,319	↑ 0.5
SC $F = 160$	2829	↓ 13.2	25,572	↑ 1.5
DC1	2379	↓ 27.0	21,460	↓ 14.9
DC2	2344	↓ 28.1	21,389	↓ 15.1
DC3	2429	↓ 25.4	22,019	↓ 12.6
Clairvoyant	1792	↓ 45.0	17,140	↓ 32.0

(b) Setting: top-$k = 1{,}000$

Strategy	MSMARCO-v1		MSMARCO-v2	
	Total (sec)	Change (%)	Total (sec)	Change (%)
Naïve	11,958	—	95,881	—
Q_k	8910	↓ 25.5	64,145	↓ 33.1
SC $F = 40$	7857	↓ 34.3	61,738	↓ 35.6
SC $F = 80$	7846	↓ 34.4	66,498	↓ 30.6
SC $F = 160$	7987	↓ 33.2	68,611	↓ 28.4
DC1	8399	↓ 29.8	55,343	↓ 42.3
DC2	7520	↓ 37.1	53,332	↓ 44.4
DC3	7936	↓ 33.6	55,562	↓ 42.1
Clairvoyant	7173	↓ 40.0	48,849	↓ 49.1

a clear balance between the total time taken and the value of F, with $F = 80$ performing best on the smaller collection, and $F = 40$ on the larger collection; we explore this trade-off in more detail shortly.

The three *dynamic caching* mechanisms demonstrate even further time savings – up to 87 minutes on MSMARCO-v1, and 709 minutes on MSMARCO-v2. Both DC1 and DC2 slightly outperform DC3, indicating that the cost of cache access (even across all 1, 2, and 3 term subqueries) is outweighed by the subsequent reduction in total processing cost once a suitable threshold is found; these trends hold irrespective of both the query processing algorithm, and the value of k applied.

Table 3: Total time (including all pre-processing operations) and relative latency change over the Naïve baseline when executing all 6.76 million queries on both `MSMARCO-v1` and `MSMARCO-v2` indexes, using the VBMW algorithm.

(a) Setting: top-$k = 10$

Strategy	MSMARCO-v1		MSMARCO-v2	
	Total (sec)	Change (%)	Total (sec)	Change (%)
Naïve	3492	—	19,597	—
Q_k	3155	↓ 9.7	18,793	↓ 4.1
SC $F = 40$	2790	↓ 20.1	19,248	↓ 1.8
SC $F = 80$	2902	↓ 16.9	19,411	↓ 1.0
SC $F = 160$	3044	↓ 12.8	19,675	↑ 0.4
DC1	2519	↓ 27.9	17,716	↓ 9.6
DC2	2484	↓ 28.9	17,546	↓ 10.5
DC3	2582	↓ 26.1	17,478	↓ 10.8
Clairvoyant	1850	↓ 47.0	13,006	↓ 33.6

(b) Setting: top-$k = 1{,}000$

Strategy	MSMARCO-v1		MSMARCO-v2	
	Total (sec)	Change (%)	Total (sec)	Change (%)
Naïve	14,213	—	65,967	—
Q_k	10,911	↓ 23.2	57,432	↓ 12.9
SC $F = 40$	9435	↓ 33.6	51,296	↓ 22.2
SC $F = 80$	9383	↓ 34.0	53,380	↓ 19.1
SC $F = 160$	9496	↓ 33.2	55,656	↓ 15.6
DC1	9021	↓ 36.5	47,296	↓ 28.3
DC2	9045	↓ 36.4	46,257	↓ 29.9
DC3	9541	↓ 32.9	48,781	↓ 26.1
Clairvoyant	8004	↓ 43.7	36,516	↓ 44.6

Overall, these results are encouraging, especially for larger values of k, with latency reductions generally ranging between 15% to 42% of the baseline latency. One exception is the larger `MSMARCO-v2` collection with $k = 10$, where only modest improvements were observed. We believe this is due to the scaling behavior of dynamic pruning algorithms (Gallagher et al., 2025). In particular, this setting (large collection, small k value) is more favorable to dynamic pruning algorithms than others which, in turn, means that the benefits of batch processing are eroded. Contrasting these findings to the Clairvoyant algorithm shows that there may still be significant room for improvement, although this idealistic baseline does not incur any cost for *finding* the optimal threshold, something that is impossible in practice.

Table 4: Space overhead, measured in MiB, for the given algorithms on the larger ORCAS batch. The Q_k algorithm assumes storage of three k settings for each term in the index; the other algorithms are sensitive to only the batch itself. Note that all variations of the DC algorithm use the same caching scheme, and thus have the same overhead.

Q_k	Static			Dynamic
	$F = 40$	$F = 80$	$F = 160$	
30–190	161	144	130	197

5.2 Space Overheads

Table 4 reports the maximum space consumption of each strategy in MiB. Note that the space usage of the Q_k strategy depends on the size of the vocabulary; we have assumed that the kth highest score is stored for three values of k, using four bytes each. For the Static and Dynamic algorithms, the cache grows in terms of the size of $\mathcal{B}$, rather than the index itself. Given that the indexes are 837 MiB and 22.7 GiB for MSMARCO-v1 and MSMARCO-v2, respectively, the relative overhead of the threshold cache is much higher on the smaller index, approaching 25% of the total index size. On the larger collection, however, this overhead is less than 1%, making it an attractive mechanism for large-scale batch processing tasks. We acknowledge that this space could also be used for alternative enhancements – such as caching postings lists, or pre-computing results for specific sub-queries – and we plan to investigate alternative uses of this space in future work.

5.3 Static Caching Costs

Looking further at the cost of each component of the SC algorithm reveals that the most significant component is, as expected, the *scoring* stage. Generating all sub-queries for the 6.76 million query batch was observed to take around 30 seconds; filtering out all sub-queries occurring less than F times takes about 1 second; and once the thresholds are available, associating them back to their parent queries in $\mathcal{B}$ takes around 1 second. However, the sub-query *scoring* operation was observed to take up to 229 seconds for MSMARCO-v1, and 1165 seconds for MSMARCO-v2, using VBMW under the most expensive setting of $F = 40$ (with $k = 1{,}000$). On the other end of that spectrum, sub-query scoring with $F = 160$ takes a total of 40 seconds on MSMARCO-v1, and 374 seconds on MSMARCO-v2. Yet, the total run-time shown in Table 3 demonstrates that savings arising from pre-processing are not recouped during the final query processing phase, indicating the absence of useful thresholds.

Based on these results, a clear avenue for improvement is to improve the sub-query selection process. For example, many queries contain common term pairs such as *"how do"* or *"who is,"* yet these pairs are unlikely to yield high Θ_k values, and may not accelerate querying at all, making their computation a waste of resources. On the other hand, the sub-query scoring only takes up to 3% of the total processing time, and given the large gap between the aspirational clairvoyant approach and the static caching algorithm, more intelligent (and perhaps resource intensive) sub-query selection could yield much better outcomes.

5.4 Dynamic Caching Hit Analysis

Next, we examine the cache behavior of the DC algorithms in more detail to provide a better understanding of their performance.

Since $\mathcal{B}$ contains only unique queries, it is impossible for the DC algorithms to yield a cache hit for any single term query; as such, the first 211,273 (single term) queries are cache misses (see Figure 3).

Both DC1 and DC2 use the same sub-query lookup sequence, differing only once a cache hit occurs; across the tested batch, only 1,693 queries (ignoring the aforementioned singles) were not associated with a threshold value for these heuristics. In other words, more than 97% of the roughly 6.7 million queries started with a non-zero value of θ under either DC1 or DC2, explaining their strong performance. On the other hand, DC3, which searches for only $|q| - 1$ length sub-queries, had 607,224 misses (about 9%), again ignoring the singles. This high cache hit ratio can be explained by multiple factors. Firstly, recall that the batch is sorted by query length, and then lexicographically within length groups. English language is known to be Zipfian in nature, meaning that common terms occur very frequently. Thus, a common term appearing as a single term query in the batch could result in a large number of cache hits in subsequent queries containing that term, as could a common pair, and so on. Secondly, the ORCAS query log is biased towards popular queries, as only queries that were submitted by multiple unique users *and* led to clickthroughs were retained in the log.

In contrast to the DC algorithms, the SC algorithm does not require any cache look-ups during query processing, with all sub-query association done during pre-processing. However, there is a clear benefit in caching the results of queries during processing; since all queries in $\mathcal{B}$ need to be processed, storing the resulting thresholds and consulting a cache – even considering the cost of look-ups themselves – seems to be a worthwhile optimization.

5.5 Batch Size Effects

In the previous experiment, query batches were made up of all available data. Our next experiment aims to quantify the influence of different batch sizes on the overall performance of the algorithms presented. In order to measure this, we generated random subsets of the ORCAS batch of various sizes, without replacement, and ran the best in-class algorithms from Section 5.1 on these subsets. Table 5 reports the outcomes. As expected, the proposed batch processing algorithms tend to perform similarly to the baseline approach for smaller batches, as there are fewer overlapping query terms to exploit. Interestingly, however, there are still clear benefits to applying batch processing algorithms on inputs that are as small as $|\mathcal{B}| = 10^4$ (with savings observed of up to 50%). From an operational perspective, however, batch processing is unlikely to be a worthwhile optimization for batches of this size, as the total computation required to process them is somewhat negligible; even processing a batch of 10^4 queries on the larger MSMARCO-v2 collection takes just over one and a half minutes with the naïve approach. On the other hand, even when batch processing does not largely help, it does not significantly *hurt* either, making it a reasonably safe optimization to incorporate into a large-scale querying system.

We remark that the notion of drawing *random* subsets of queries is rather strict for the batch processing algorithms as their performance depends on occurrences of overlapping sub-queries. Thus, it is possible that batch processing can potentially be a helpful

Table 5: Total time (including all pre-processing operations), in seconds, to process batches of varying sizes over both collections and settings of k. This experiment uses VBMW processing.

(a) Setting: $k = 10$

Strategy	MSMARCO-v1					MSMARCO-v2				
	10^3	10^4	10^5	10^6	All	10^3	10^4	10^5	10^6	All
Naïve	0.5	5	51	511	3492	5.8	33	284	2869	19,597
SC $F = 40$	1.0	10	106	482	2790	2.9	29	297	2925	19,248
DC2	0.5	5	52	472	2484	2.9	30	298	2892	17,546

(b) Setting: $k = 1{,}000$

Strategy	MSMARCO-v1					MSMARCO-v2				
	10^3	10^4	10^5	10^6	All	10^3	10^4	10^5	10^6	All
Naïve	2.0	21	211	2004	14,213	9.6	96	972	9748	65,967
SC $F = 40$	1.9	19	162	1455	9435	9.7	96	941	8615	51,296
DC2	2.0	21	207	1767	9045	9.8	98	977	8838	46,257

optimization on smaller batches, especially when those batches are likely to contain related queries, such as for processing a set of query variations (Benham et al., 2019) or topically clustered batches. We look forward to exploring this problem in future work.

5.6 Parallel Processing

In our prior commentary, we have assumed that the batch processing engine has no latency requirements, and that processing can proceed on a single CPU. However, we note that there may be situations where the total end-to-end time to process the batch may still need to be as fast as possible. To this end, we also explore how each top-k batch processing method scales in the multi-threaded processing scenario, with the aim of completing the entire query batch in the minimal end-to-end time.

Figure 4 shows the total time taken to process the batch over both indexes, using VBMW with $k = 1{,}000$. It is evident that the parallel processing optimizations do not lead to any adverse behavior, with almost perfect parallelism across all tested algorithms. In turn, this means the computational savings achieved from our enhanced batch query processing methods are not eroded under parallel processing, making them a suitable choice when the overall batch latency must be optimized while still saving on overall computing cost, making these optimizations quite practical for real-world systems.

5.7 Generalizability

Our final experiment aims to verify the generalizability of our batch processing approaches to other query batches. To build a new batch, we used the 9.2 million training queries from

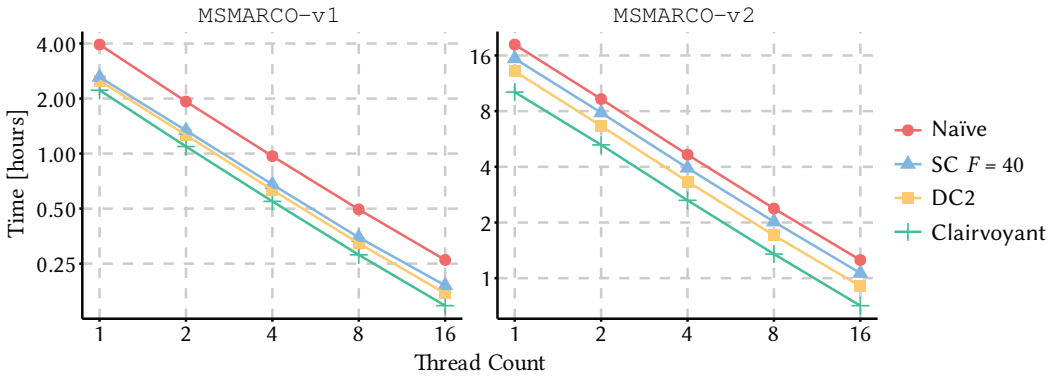


Figure 4: The total latency taken to process the entire batch, in hours, as the amount of processing threads increases. Both experiments are for k = 1,000 with VBMW. All approaches scale almost perfectly as threads are added – note the logarithmic axes.

Table 6: Total time (including all pre-processing operations), in seconds, to process two different batches over MSMARCO-v2 with VBMW and k = 1,000.

Strategy	ORCAS		MSM-WS	
	Total (sec)	Change (%)	Total (sec)	Change (%)
Naïve	65,967	—	63,677	—
Q_k	57,432	↓ 12.9	59,767	↓ 6.1
SC $F = 40$	51,296	↓ 22.2	59,054	↓ 7.3
SC $F = 80$	53,380	↓ 19.1	59,627	↓ 6.4
SC $F = 160$	55,656	↓ 15.6	60,459	↓ 5.1
DC1	47,296	↓ 28.3	58,662	↓ 7.9
DC2	46,257	↓ 29.9	58,494	↓ 8.1
DC3	48,781	↓ 26.1	61,753	↓ 3.0
Clairvoyant	36,516	↓ 44.6	44,096	↓ 30.8

the MSMARCO Web Search collection (Chen et al., 2024) as a starting point.[2] We then filtered out any non-English queries, normalized all queries, and removed duplicates or queries with out-of-vocabulary terms, following the original experimental setup (Mackenzie and Moffat, 2023). This resulted in a new batch containing 3,832,510 unique queries. While this batch is smaller than the ORCAS batch used in prior experiments, it has a longer average query length (3.7 vs 3.2 terms, respectively, see Figure 3). Repeating our experiments on this new batch showed similar trends; the Clairvoyant algorithm had a 30% to 44% improvement over the Naïve baseline, with both the DC and SC algorithms following behind with a modest 3% to 8% improvement. This result indicates that while our batch processing improvements

2. Note that, despite the name, the MSM-WS query set is not derived from (nor is aligned with) the MSMARCO corpora used in our experimentation – rather, it is based on the ClueWeb22 corpus.

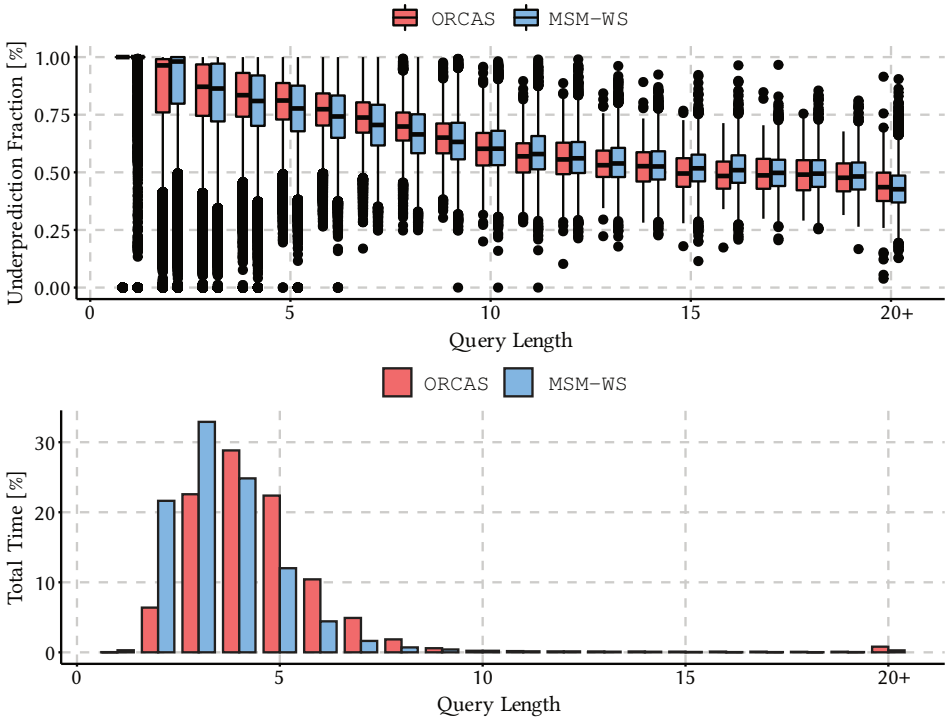


Figure 5: Top: The per-query threshold underprediction distribution across both logs, bucketed by query length. Bottom: The proportion of time spent processing queries across both logs, bucketed by query length. These figures use the same settings as in Table 6: VBMW processing with $k = 1{,}000$ on MSMARCO-v2.

do generalize to other query loads, the observed improvements almost certainly depend on the size and properties of the batch, and the corpus, itself.

To further illustrate this, Figure 5 plots the *underprediction fraction* (Petri et al., 2019) of the SC algorithm using $F = 40$. The underprediction fraction represents the difference, as a percentage, of the threshold that was applied to each query against the optimal threshold, meaning that a value of 1.0 represents a perfect initial threshold. We also plotted the distribution of time spent on processing queries of different lengths across the batch. We can observe that, as the length of the query increases, the estimations become worse. This is intuitive, because longer queries result in higher Θ_k scores, but we only use up to three-term sub-queries for initial threshold estimation. More importantly, we see that the underprediction fractions on the ORCAS log tend to be higher than on the MSM-WS log, especially for queries containing 3 to 9 terms which make up over 70% of the total batch processing time. The higher the underprediction fraction, the more savings accrue.

6 Discussion and Future Work

Our experiments demonstrated that the offline method is generally outperformed by its dynamic, online counterparts. Furthermore, comparing the savings of both approaches to the Clairvoyant yardstick shows that there is still room for improvement.

The key benefit of pre-processing term associations offline is that there is no need to maintain a cache on-the-fly; each query is already associated with the best possible threshold prior to querying, so no enumeration (or hashing) of sub-queries is required when processing the batch. However, as discussed in Section 5.3, there is also a delicate balance between the cost of pre-computing the thresholds for candidate sub-queries, and the savings realized from these thresholds during processing. In this work, we have only applied simple frequency-based cut-offs to determine whether a sub-query is included or not. More sophisticated regimes, including dynamic regimes based on features such as the length of a query, or the estimated utility of a given sub-query, could lead to better trade-offs than those presented here. We leave this investigation to future work.

One promising direction is to adapt both approaches into a *hybrid* algorithm, using both offline and online computation to accelerate the batch. It is unclear, however, whether these mechanisms are complementary (and hence, additive in their effects (Mackenzie and Moffat, 2020; Armstrong et al., 2009)), or if they are simply optimizing the same underlying inefficiencies.

Another possibility for closing this gap is to apply threshold *prediction* algorithms (Petri et al., 2019; Mallia et al., 2020), which use other features to predict the terminal value of the heap prior to processing each query. However, predictive algorithms come at the risk of *overestimation* which can harm effectiveness, something we did not consider here.

Another assumption made in this work is that the value of k – the number of documents to retrieve – is fixed across the entire query batch. However, there could be situations where each individual query is parameterized by a different value of k. In turn, this means that only thresholds computed for sub-queries to depth $k' > k$ can be utilized for rank-safe retrieval, making the batch processing task much more difficult. We leave this interesting idea for future investigation.

Finally, it would be worthwhile to revisit this research in terms of alternative scoring mechanisms, such as *learned sparse retrieval* (Yates et al., 2024), alternative indexing frameworks, like *approximate nearest neighbor* search over dense indexes (Bruch, 2024), or different query processing contexts, like *query-by-document*, which is used for building *corpus graphs* (Dunn et al., 2025).

7 Conclusion

We explored the problem of batch query processing for disjunctive top-k retrieval algorithms. We proposed two baseline methods, as well as two competitive approaches, all based on threshold estimation. These techniques reduce the overall cost of processing a large batch of queries. Our experiments demonstrated that a *dynamic caching* approach, based on the work of Yafay and Altingovde (2019), provided the most benefit, reducing the total cost of processing by as much as 44% – almost halving the processing time – with a small associated space overhead. We also found that these improvements translate to parallel

processing, allowing batches to be processed faster while still retaining cost reductions over naïve query-by-query processing. With many open problems yet to be answered, we hope that our work can motivate further studies into the problem of batch processing, especially for top-k retrieval.

Acknowledgments and Disclosure of Funding

The authors thank Alistair Moffat, as well as the anonymous referees, for useful conversations and suggestions. The second author was supported by a Google Research Scholar grant. In the interest of reproducibility, all experimental code is available at the following URL: `https://github.com/JMMackenzie/disbatch`.

References

T. G. Armstrong, A. Moffat, W. Webber, and J. Zobel. Improvements that don't add up: Ad-Hoc retrieval results since 1998. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 601–610, 2009.

X. Bai, I. Arapakis, B. B. Cambazoglu, and A. Freire. Understanding and leveraging the impact of response latency on user behaviour in web search. *ACM Trans. on Information Systems*, 36(2):1–42, 2017.

P. Bajaj, D. Campos, N. Craswell, L. Deng, J. Gao, X. Liu, R. Majumder, A. McNamara, B. Mitra, T. Nguyen, M. Rosenberg, X. Song, A. Stoica, S. Tiwary, and T. Wang. MS MARCO: A human generated MAchine Reading COmprehension dataset. *arXiv:1611.09268v3*, 2018.

R. Benham, J. Mackenzie, A. Moffat, and J. S. Culpepper. Boosting search performance using query variations. *ACM Trans. on Information Systems*, 37(4):41.1–41.25, 2019.

R. Blanco, M. Catena, and N. Tonellotto. Exploiting green energy to reduce the operational costs of multi-center web search engines. In *Proc. Conf. on the World Wide Web (WWW)*, pages 1237–1247, 2016.

A. Z. Broder, D. Carmel, M. Herscovici, A. Soffer, and J. Zien. Efficient query evaluation using a two-level retrieval process. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 426–434, 2003.

S. Bruch. *Foundations of Vector Retrieval*. Springer, 2024. ISBN 9783031551826.

C. Buckley and A. F. Lewit. Optimization of inverted vector searches. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 97–110, 1985.

M. Catena, C. Macdonald, and N. Tonellotto. Load-sensitive CPU power management for web search engines. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 751–754, 2016.

Q. Chen, X. Geng, C. Rosset, C. Buractaon, J. Lu, T. Shen, K. Zhou, C. Xiong, Y. Gong, P. Bennett, N. Craswell, X. Xie, F. Yang, B. Tower, N. Rao, A. Dong, W. Jiang, Z. Liu, M. Li, C. Liu, Z. Li, R. Majumder, J. Neville, A. Oakley, K. M. Risvik, H. V. Simhadri, M. Varma, Y. Wang, L. Yang, M. Yang, and C. Zhang. MS MARCO Web Search: A large-scale information-rich web dataset with millions of real click labels. In *Proc. Conf. on the World Wide Web (WWW)*, pages 292–301, 2024.

F. M. Choudhury, J. S. Culpepper, Z. Bao, and T. Sellis. Batch processing of top-k spatial-textual queries. *ACM Trans. on Spatial Algorithms and Systems*, 3(4):13.1–13.40, 2018.

G. Chowdhury. An agenda for green information retrieval research. *Information Processing & Management*, 48(6):1067–1077, 2012.

G. V. Cormack, M. D. Smucker, and C. L. A. Clarke. Efficient and effective spam filtering and re-ranking for large web datasets. *Information Retrieval*, 14(5):441–465, October 2011.

M. Crane, A. Trotman, and R. O'Keefe. Maintaining discriminatory power in quantized indexes. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 1221–1224, 2013.

M. Crane, J. S. Culpepper, J. Lin, J. Mackenzie, and A. Trotman. A comparison of Document-at-a-Time and Score-at-a-Time query evaluation. In *Proc. ACM Int. Conf. on Web Search and Data Mining (WSDM)*, pages 201–210, 2017.

N. Craswell, D. Campos, B. Mitra, E. Yilmaz, and B. Billerbeck. ORCAS: 20 million clicked query-document pairs for analyzing search. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 2983–2989, 2020.

L. L. S. de Carvalho, E. S. de Moura, C. M. Daoud, and A. S. da Silva. Heuristics to improve the BMW method and its variants. *Journ. Information and Data Management*, 6(3):178–191, 2015.

L. Dhulipala, I. Kabiljo, B. Karrer, G. Ottaviano, S. Pupyrev, and A. Shalita. Compressing graphs and indexes with recursive graph bisection. In *Proc. Conf. on Knowledge Discovery and Data Mining (KDD)*, pages 1535–1544, 2016.

S. Ding and T. Suel. Faster top-k document retrieval using block-max indexes. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 993–1002, 2011.

S. Ding, J. Attenberg, R. Baeza-Yates, and T. Suel. Batch query processing for web search engines. In *Proc. ACM Int. Conf. on Web Search and Data Mining (WSDM)*, pages 137–146, 2011.

L. Dunn, L. Gallagher, and J. Mackenzie. Approximate bag-of-words top-k corpus graphs. In *Proc. European Conf. on Information Retrieval (ECIR)*, pages 174–182, 2025.

M. Fontoura, V. Josifovski, J. Liu, S. Venkatesan, X. Zhu, and J. Zien. Evaluation strategies for top-k queries over memory-resident inverted indexes. *Proc. Conf. on Very Large Databases (VLDB)*, 4(12):1213–1224, 2011.

L. Gallagher, R-C. Chen, R. Blanco, and J. S. Culpepper. Joint optimization of cascade ranking models. In *Proc. ACM Int. Conf. on Web Search and Data Mining (WSDM)*, pages 15–23, 2019.

L. Gallagher, J. Mackenzie, and A. Moffat. Empirical asymptotic growth of dynamic pruning mechanisms. In *Proc. Int. ACM SIGIR Conference on Information Retrieval in the Asia Pacific (SIGIR-AP)*, pages 325–330, 2025.

C. Kamphuis, A. P. de Vries, L. Boytsov, and J. Lin. Which BM25 do you mean? A large-scale reproducibility study of scoring variants. In *Proc. European Conf. on Information Retrieval (ECIR)*, pages 28–34, 2020.

A. Kane and F. W. Tompa. Split-lists and initial thresholds for WAND-based search. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 877–880, 2018.

E. Kayaaslan, B. B. Cambazoglu, R. Blanco, F. P. Junqueira, and C. Aykanat. Energy-price-driven query processing in multi-center web search engines. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 983–992, 2011.

D. Lemire and L. Boytsov. Decoding billions of integers per second through vectorization. *Software Practice & Experience*, 41(1):1–29, 2015.

J. Lin, J. Mackenzie, C. Kamphuis, C. Macdonald, A. Mallia, M. Siedlaczek, A. Trotman, and A. de Vries. Supporting interoperability between open-source search engines with the common index file format. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 2149–2152, 2020.

J. Mackenzie and A. Moffat. Examining the additivity of top-k query processing innovations. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 1085–1094, 2020.

J. Mackenzie and A. Moffat. Index-based batch query processing revisited. In *Proc. European Conf. on Information Retrieval (ECIR)*, pages 86–100, 2023.

J. Mackenzie, M. Petri, and A. Moffat. Tradeoff options for bipartite graph partitioning. *IEEE Trans. on Knowledge and Data Engineering*, 35(8):8644–8657, 2022.

A. Mallia, G. Ottaviano, E. Porciani, N. Tonellotto, and R. Venturini. Faster BlockMax WAND with variable-sized blocks. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 625–634, 2017.

A. Mallia, M. Siedlaczek, J. Mackenzie, and T. Suel. PISA: Performant indexes and search for academia. In *Proc. OSIRRC at SIGIR 2019*, pages 50–56, 2019a.

A. Mallia, M. Siedlaczek, and T. Suel. An experimental study of index compression and DAAT query processing methods. In *Proc. European Conf. on Information Retrieval (ECIR)*, pages 353–368, 2019b.

A. Mallia, M. Siedlaczek, M. Sun, and T. Suel. A comparison of top-k threshold estimation techniques for disjunctive query processing. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 2141–2144, 2020.

L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web, 1999. URL `http://ilpubs.stanford.edu:8090/422/`. Technical Report.

M. Petri, J. S. Culpepper, and A. Moffat. Exploring the magic of WAND. In *Proc. Australasian Document Computing Symp. (ADCS)*, pages 58–65, 2013.

M. Petri, A. Moffat, J. Mackenzie, J. S. Culpepper, and D. Beck. Accelerated query processing via similarity score prediction. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 485–494, 2019.

S. E. Robertson and H. Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations & Trends in Information Retrieval*, 3:333–389, 2009.

H. Scells, S. Zhuang, and G. Zuccon. Reduce, reuse, recycle: Green information retrieval research. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 2825–2837, 2022.

E. Schurman and J. Brutlag. Performance related changes and their user impact. Velocity, 2009.

D. Shan, S. Ding, J. He, H. Yan, and X. Li. Optimized top-k processing with global page scores on block-max indexes. In *Proc. ACM Int. Conf. on Web Search and Data Mining (WSDM)*, pages 423–432, 2012.

L. H. Thiel and H. S. Heaps. Program design for retrospective searches on large data bases. *Information Storage and Retrieval*, 8(1):1–20, 1972.

N. Tonellotto, C. Macdonald, and I. Ounis. Efficient query processing for scalable web search. *Foundations & Trends in Information Retrieval*, 12(4-5):319–500, 2018.

H. R. Turtle and J. Flood. Query evaluation: Strategies and optimizations. *Information Processing & Management*, 31(6):831–850, 1995.

L. Wang, J. Lin, and D. Metzler. Learning to efficiently rank. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 138–145, 2010.

E. Yafay and I. S. Altingovde. Caching scores for faster query processing with dynamic pruning in search engines. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 2457–2460, 2019.

P. Yang, H. Fang, and J. Lin. Anserini: Reproducible ranking baselines using Lucene. *Journal of Information Quality*, 10(4):1–20, 2018.

A. Yates, C. Lassance, S. MacAvaney, T. Nguyen, and Y. Lei. Neural lexical search with learned sparse retrieval. In *Proc. Int. ACM SIGIR Conference on Information Retrieval in the Asia Pacific (SIGIR-AP)*, pages 303–306, 2024.

J. Zobel and A. Moffat. Inverted files for text search engines. *ACM Computing Surveys*, 38 (2):6.1–6.56, 2006.

G. Zuccon, H. Scells, and S. Zhuang. Beyond CO2 emissions: The overlooked impact of water consumption of information retrieval models. In *Proc. Int. Conf. on Theory of Information Retrieval (ICTIR)*, pages 283–289, 2023.

Information Retrieval Research 2 (2026) 59-112 Submitted 12/25; Revised 05/26; Published 06/26

Much Ado about Accessibility: An Exploration of Online Content Accessibility from an Autism-Informed Perspective

Hrishita Chakrabarti H.CHAKRABARTI@TUDELFT.NL
Delft University of Technology
Delft, Netherlands

Maria Soledad Pera M.S.PERA@TUDELFT.NL
Delft University of Technology
Delft, Netherlands

Editor: Haiming Liu

Abstract

Autism Spectrum Disorder is a neuro-developmental condition that may affect the way a person learns, behaves, and interacts with their environment. Autistic individuals are known to often rely on mainstream information access systems, such as search engines, as their initial point for information discovery. Despite their active online information-seeking practices, autistic individuals struggle to extract information from the resources they encounter, leading to frustration. This raises the question of whether these systems fit the needs and expectations of this population. In this work, we empirically explore how different mainstream information access systems respond to autistic users' inquiries through the lens of web content accessibility. Specifically, we focus on three text-based perspectives known to impact autistic individuals' information seeking – structure, readability, and concreteness (tangibility of terminology) of web content. For our analysis, we leverage established accessibility guidelines for autistic individuals to create accessibility profiles for Google, Bing, ChatGPT, and Gemini, providing an overview of various aspects of each system's responses to common inquiries of autistic individuals. Given the influence of query formulation on produced responses, we also investigate the impact of explicitly informing the system of the user's condition on the accessibility profile of the considered systems. Our findings reveal potential accessibility limitations of mainstream information access systems when responding to autistic users' inquiries, more so if their information need is related to the searchers' condition. Furthermore, including a diagnosis-disclosing phrase in autistic users' inquiries results in responses that are even less accessible to autistic individuals. Based on our findings, we reflect on autistic users' access to information and the role of information access technologies in supporting their information-seeking process.

Keywords: generative models, text analysis, autism, web accessibility, search engines, LLMs

1 Introduction

Autism Spectrum Disorder (**ASD**) is a neuro-developmental condition that affects how a person interacts and communicates with the people and environment around them (CDC, 2024b). The abilities of autistic individuals[1] can vary immensely, i.e., individuals' abilities

1. There is an ongoing debate on how Autism must be described. Here, we use "identity-first" (e.g., autistic individuals or autistic users) instead of "person-first" language (e.g., individual with Autism or user with Autism) based on the preferences of autistic individuals (Kenny et al., 2016; Bottema-Beutel et al., 2021).

 WWW: https://irrj.org DOI: 10.54195/irrj.25297

lie on a *spectrum* with an autistic person's needs potentially changing over time (WHO, 2023). Despite their heterogeneity, common struggles include issues with memory, attention, stimulus sensitivity, and language and visual comprehension (APA, 2022; CDC, 2024a). These conditions often hinder autistic users from socialising and carrying out their daily activities, leading to loneliness and poor quality of life (Locke et al., 2010; Lin and Huang, 2019).

Autistic individuals display a strong affinity for technology–especially that aiming to support their daily activities (Lin et al., 2013; Putnam and Chong, 2008). They also have a prominent presence on online platforms, which they use for various purposes, including establishing and maintaining relationships through social media platforms as well as searching for information related to their interests (Bosseler and Massaro, 2003; Kim et al., 2020; Gibson and Hanson-Baldauf, 2019). For the latter, they often turn to popular information access systems (**IASs**), such as search engines (**SEs**) like Google and Bing, as the portal to online information (Gibson and Hanson-Baldauf, 2019; Jang et al., 2024). However, they sometimes struggle with extracting information from the online resources they come across (e.g., Eraslan et al., 2017; Friedman and Bryen, 2008). This is known to stem from various factors; of note, autistic users find it difficult to locate relevant components of information when a website is too visually complex due to autistic individuals' attention issues and challenges with visual comprehension (White and Abou-Zahra, 2024; Eraslan et al., 2017). Additionally, the phrasing of textual content in an online source can impact an autistic person's comprehension of the content. Vague metaphors and "fancy" language, used to make content engaging, can sometimes end up confusing an autistic person (White and Abou-Zahra, 2024; Yaneva et al., 2015).

Autistic individuals are increasingly turning to generative AI-based IASs for information seeking, i.e., LLM-agents like ChatGPT and Gemini (White, 2024), (Jang et al., 2024; Spengler, 2023), which leverage Large Language Models (**LLMs**) for response generation. This is due to the generative capabilities of these agents to personalise the response based on an autistic user's specific cognitive processing styles (Jang et al., 2024; Spengler, 2023). LLM agents also remove the issue of crowded websites but respond to users' inquiries in long passages of text, which may lead to information overload, inciting challenging behaviours or meltdowns in autistic users, which hinders autistic individuals from utilising the agent's answer for their information discovery (White and Abou-Zahra, 2024). These challenges highlight the specific needs and expectations of autistic users when seeking information online. However, there is limited understanding of how well popular IASs accommodate for autistic users' distinct requirements while catering to the user group's information needs. This prompts the need to examine whether popular IASs respond to autistic individuals' inquiries in a manner suitable for autistic users.

Inspired by the Web Accessibility Initiative (**WAI**) of the World Wide Web Consortium (**W3C**), which aims to ensure that the web content users consume accommodates their distinct abilities (W3C, 2019), in this work we conduct an empirical exploration to gauge the extent to which responses by four popular commercial IASs—Google, Bing, ChatGPT, and Gemini—for autistic individuals' inquiries align with established Web Content Accessibility Guidelines (**WCAG**) for autistic individuals. Considering how an online inquiry is formulated biases the response elicited from an IAS (Alaofi et al., 2022; Murgia et al., 2023a; Schmidt et al., 2023), we further investigate the impact of explicitly stating the

user's condition in autistic users' inquiry on the accessibility of the response. To guide our exploration, we pose two research questions:

- **RQ1:** Are the responses of mainstream SEs and LLM agents accessible to autistic users?
- **RQ2:** Are the responses of SEs and LLM agents more accessible when diagnosis is disclosed in a query?

For our exploration, we rely on the WCAG proposed by the Cognitive and Learning Disabilities Accessibility Task Force (**COGA**) – a task force under WAI's Accessibility Guidelines Working Group, created to understand and address the challenges faced by autistic users (amongst other user groups with cognitive and learning disabilities) when exposed to content on the web (W3C, 2021). We scope our investigations to the *textual content* of IAS responses and create *accessibility profiles*. Leveraging the outcomes of empirical studies that validate the applicability of COGA's guidelines (e.g., Yaneva et al., 2015; Eraslan et al., 2017), an accessibility profile provides an overview of various text-based aspects of an IAS response to user inquiries that are known to impact autistic individuals' information seeking. Particularly, we focus on aspects related to three broad perspectives of text accessibility:

1. *Structure*, i.e., how concise the passages are, as guidelines suggest that text should be written in short paragraphs or a list of bullet points to avoid inciting information overload in autistic users (Britto and Pizzolato, 2016; Seeman and Cooper, 2024),
2. *Readability* or in other words language complexity, as autistic individuals prefer simple and easy-to-read sentences over complex sentences with niche jargon (Yaneva et al., 2015),
3. *Concreteness*, which refers to the tangibility of the word describing a concept (Paivio, 2013), as text accessible to autistic users should explain related concepts concretely and avoid vague metaphors as much as possible (Yaneva et al., 2015; Yaneva and Evans, 2015).

To ascertain if the accessibility of IAS responses is inherent to general algorithmic behaviour intrinsic to an IAS or if it changes depending on a particular user group and their inquiries, we also create accessibility profiles of the considered IASs based on their responses for inquiries formulated by the general population, which we treat as the control group. Due to the scarcity of public logs capturing IAS responses to user queries, we generate the accessibility profiles for the considered IASs based on responses elicited using query samples that reflect the common information needs of autistic users and the general population, respectively. Query samples from a general population are commonly available for research purposes; in our case, we use a random sample from Yahoo! Search Query Tiny sample dataset (Yahoo!, 2009). Unfortunately, queries from autistic users are generally not shared or part of publicly available datasets. Consequently, we anchor our exploration on a newly generated sample of synthetic queries that represent the information needs of autistic individuals. For this, informed by similar studies conducted for other non-traditional user groups, such as those suffering from mental health disorders for whom data is also scarce

(e.g., De Choudhury et al., 2016; Ríssola et al., 2022; Milton and Pera, 2023), we turn to Reddit to collect posts made by self-reported autistic users on various forums dedicated to different topics related to the target population. We extract popular noun phrases from these posts to use as *proxy queries* related to topics of interest for autistic individuals that also capture the common vocabulary and phrasing used by the target population.

Findings reveal that all the considered IASs respond with verbose, hard-to-read, and abstract texts for autistic individuals' queries compared to the responses for inquiries initiated by the general population. Moreover, when an autistic user's inquiry relates to ASD, IAS responses are much harder to read than when the inquiry pertains to other topics. Furthermore, we find that explicitly informing the IAS that the inquiry is from an autistic user results in more verbose responses, which are also more abstract in nature, compared to when the systems are unaware of the user's condition. We do, however, observe that Google responses are considerably easier to read when an autistic user's query is rephrased to explicitly state the user's condition. Furthermore, IAS responses when informed of the user's condition, although less aligned with autistic individuals' accessibility requirements on average, are more consistent in their structure, readability, and concreteness across queries.

The main contributions of this work are (i) generating accessibility profiles of IAS responses based on textual features that impact web content accessibility for autistic users during search; and (ii) providing empirical evidence that in terms of text accessibility, the considered IASs respond differently depending on the user group and their inquiries, with IAS responses being less aligned with autistic individuals' accessibility requirements when the inquiry pertains to autistic individuals' information needs compared to when the inquiry comes from the general population. This raises concerns about potential issues in autistic individuals' access to information through online mediums—a direct infringement of their fundamental human rights (United Nations, 1948)—evincing the need to adapt popular IASs to better cater to the needs and expectations of autistic people. To support research in this direction, the experimental setup we use to generate accessibility profiles can be extended to examine IAS responses from other perspectives of web content accessibility to realise other potential accessibility limitations of online IASs. For reproducibility purposes and to facilitate further analysis to understand information accessibility for not only autistic people but also other vulnerable and underserved user groups, we share our code at: `https://github.com/HrishitaC/much-ado-about-accessibility`.

2 Related Work

We discuss below related literature that offers context and informs the experimental designs of our empirical investigation into the accessibility of responses provided by popular IASs to autistic users.

Autism Spectrum Disorder. ASD is a neuro-developmental condition that causes affected individuals to communicate and behave in ways that are different from most people (CDC, 2024b). Autistic people can begin showing symptoms as early as the age of 2 years, but most individuals are diagnosed much later in their lives (Nyrenius et al., 2023; Lai and Baron-Cohen, 2015). Currently, it is estimated that 1 in every 100 children is autistic (WHO, 2023), with the condition more prevalent in boys than girls (Zeidan et al., 2022). Increased awareness and changes in diagnostic criteria have resulted in a significant rise

in autism diagnoses over the past few years (Zeidan et al., 2022; Russell et al., 2022), yet ASD remains largely underdiagnosed (O'Nions et al., 2023; WHO, 2023), especially among women (Cary et al., 2023; Driver and Chester, 2021).

The abilities of autistic individuals are described to lie on a spectrum, with the symptoms and effects of the condition varying from person to person and evolving with age (CDC, 2024a; WHO, 2023). Typically, an autism diagnosis is categorised into 3 levels depending on the extent of support the affected individual requires for their daily activities, where level 1 Autism indicates low-level support, primarily with social communication, while level 3 Autism implies that the individual requires substantial support to carry out even day-to-day activities (APA, 2022). Despite the variation in their abilities, most autistic people face issues with reading comprehension, including difficulty processing long, complex sentences and understanding figurative language and abstract words (O'Connor and Klein, 2004; Frith and Snowling, 1983; Happé, 1997). Autistic individuals also depict atypical attention patterns such as their reliance on a single sensory modality when performing a task, a phenomenon called "stimulus overselectivity" (Lovaas and Schreibman, 1971). Due to this phenomenon, autistic people often tend to be more detail-oriented and fail to see the "bigger picture", i.e., they focus on local components of information and fail to consider the global context and semantic information provided in a reading material (Happé and Frith, 2006; Ploog, 2010). Challenges with reading tasks are a common reason for lower academic and professional achievements in autistic users which in turn have long-term negative impacts on autistic people's employment rates, social relationships, and mental and physical health (Howlin and Moss, 2012) resulting in lower quality of life for autistic people compared to non-autistic individuals (Lin and Huang, 2019; Mason et al., 2018).

ASD & Technology. Autistic individuals show a natural affinity to technology to aid them in their daily life (Lin et al., 2013; Putnam and Chong, 2008). Particularly, due to their struggles with social interactions, autistic individuals have a very restricted friend circle, due to which they are reported to show higher levels of loneliness compared to the general public (Shattuck et al., 2011; Locke et al., 2010). Social media platforms have proven to be an effective tool for autistic individuals to abate this problem, with studies revealing a strong preference amongst autistic individuals to maintain their social contacts and even create new ones on these platforms (Gillespie-Lynch et al., 2014; Wang et al., 2020). Online communications provide autistic individuals with a sense of control over the conversation that they find to be lacking in a face-to-face conversation (Benford and Standen, 2009; Koteyko, 2023). Apart from maintaining their social contacts, autistic individuals also often rely on computer-based assistive technology to develop or enhance essential life skills from basic motor skills (e.g., Zheng et al., 2021; Zhao et al., 2018; Cao et al., 2025) to skills required to navigate specific social scenarios (e.g., Hartholt et al., 2019; Kandalaft et al., 2013).

Researchers have also allocated efforts in improving diagnosis and interventions for autistic individuals. For instance, autistic users' posts and their interactions on platforms like Reddit or Twitter have been used to detect linguistic and behavioural signals characteristic of autistic individuals, which can be used in detection models to facilitate early diagnosis of Autism (e.g., Kim et al., 2020; Rubio-Martín et al., 2023). In a study by Dotch et al. (2023), a qualitative analysis of the posts and comments made by self-reported autistic users on two

online forums resulted in rich insights into the personal experiences and coping mechanisms for their issues with noise sensitivity. The outcomes of the study have enabled the development of user-centred technological solutions to tackle noise sensitivity for autistic people (e.g., Park et al., 2024; Dotch, 2024). In our case, we take advantage of posts shared by self-reported autistic users on Reddit to overcome the lack of public data on online inquiries commonly asked by autistic individuals. Particularly, we extract common noun phrases from the Reddit posts to serve as synthetic queries that represent the typical information needs of autistic individuals.

The advent of LLMs has resulted in considerable advancements in diagnostic strategies for ASD (Haskins et al., 2024; Hu et al., 2024; Jiang et al., 2024). The generative capabilities of LLMs have introduced new opportunities for facilitating autistic individuals' social communication, such as providing personalised support in professional settings (Jang et al., 2024), dynamically generating feedback on the implicit emotional tone and intent in text messages (Haroon and Dogar, 2024), etc. Furthermore, researchers have leveraged LLMs to design conversational agents that can provide personalised clinical support for autistic individuals (Chu et al., 2024; Shubbar, 2025). For our study, considering the growing popularity of LLM-based chatbots like ChatGPT in autistic individuals' everyday lives (Choi et al., 2024), we examine the potential of popular LLM agents as accessible IAS for this target audience.

Regarding autistic individuals' information seeking, research efforts have resulted in the proposal of several accessibility guidelines to align web content and interfaces with the needs and expectations of autistic individuals (Britto and Pizzolato, 2016; Dattolo et al., 2016b; Raymaker et al., 2019). These guidelines, in turn, have been used to guide the development and evaluation of technological interventions to specifically facilitate autistic users' information seeking. Interventions in this area range from online tools that modify web content to enhance readability for autistic individuals (Pavlov, 2014; Eric, 2019) to online platforms that support autistic individuals information seeking in specific contexts such as tourism (Dattolo et al., 2016a; Rapp et al., 2017; Cena et al., 2020), e-learning (Judy et al., 2012; Rathod et al., 2024), medical question-answering (Chu et al., 2024), etc. Although individuals who are aware of their diagnosis do tend to utilise these resources to ease their search process (Gibson and Hanson-Baldauf, 2019), considering the low diagnosis rate of ASD around the world (WHO, 2023), many autistic individuals, unaware of their condition, may still rely on the resources a mainstream IAS would provide in response to an autistic user's inquiry. To investigate the degree to which popular IASs could support autistic users' in such scenarios, we examine the variance across different aspects of mainstream IAS responses based on text-based factors known to influence the target population's information-seeking behaviour.

ASD & Online Information Seeking. Autistic individuals frequently undertake self-directed online learning tasks on topics of their interest, but do not always trust the quality of the information available online and limit themselves to a few resources or tools that meet their expectations during a search task (Gibson and Hanson-Baldauf, 2019). The lack of trust stems from the frustration due to the unsuitable format of information on several web sources (Gibson and Hanson-Baldauf, 2019), which has been empirically proven to hinder autistic users from efficiently searching for the information they need (e.g., Eraslan et al., 2020; Yaneva et al., 2015; Rezae et al., 2020). Some researchers attribute the inefficiency

of autistic users when looking for information on the web to the difference in the attention patterns between autistic users and non-autistic users as well as the information-processing issues common in autistic individuals (APA, 2022; Williams et al., 2015). The eye scan paths of an autistic user and a non-autistic user when searching for information on a webpage were visualised in a study by Eraslan et al. (2017) highlighting the stark differences in the attention patterns of autistic users from non-autistic users when trying to extract information from a webpage. Similar studies involving tracking users' eye movements when scanning a web page revealed that autistic users scan more items on the web page, most of which were irrelevant to the question they sought to answer (Eraslan et al., 2017, 2020). Autistic adults also prefer image-based elements and feel a strong aversion to text-heavy elements (Yaneva et al., 2015). Although the image-based elements are effective in supporting autistic users' information seeking when the image is directly related to the textual content, as the presence of abstract images and icons negatively impacts the autistic users' efficiency when searching for information on a web page (Yaneva et al., 2015; Rezae et al., 2020). The impact of a web interface on autistic individuals information access is not limited to hindering autistic users ability to extract information, but can also influence other processes related to information seeking. For instance, Alzahrani et al. (2022) observed that the presence of irrelevant animations in a web interface can hinder an autistic user from formulating effective queries during search sessions.

Existing works that investigate and address the challenges autistic individuals face when seeking information online largely involve experiments in which participants complete predefined search tasks using either curated web resources (e.g., Eraslan et al., 2020; Yaneva et al., 2015) or a specific web interface (e.g., Alzahrani et al., 2022; Rezae et al., 2020). The insights gained from such controlled experiments help identify barriers that could hinder autistic individuals from finding the information they need. However, these studies do not necessarily indicate that autistic individuals face these barriers when interacting with mainstream IASs for addressing their information needs. The work by Yechiam and Yom-Tov (2021) is one of the few empirical studies based on actual browsing records of self-reported autistic users when searching for information. However, the study focuses on discerning the characteristic search behaviours of autistic users and not the extent to which the search results are accessible to autistic users, which is what we pay attention to in our study. In our work, we focus on the system and probe mainstream IAS responding to the information needs of autistic individuals, as captured in queries related to topics commonly of interest for this user group.

3 Experimental Framework

In this section, we describe the components of the experimental framework used to conduct our investigation. This work has been approved by the Human Research Ethics Committee (HREC), which is the local institutional review board for Delft University of Technology.

3.1 Information Access Systems

Considering the popularity of SE and increasing use of LLM agents amongst autistic individuals for information seeking (Yechiam and Yom-Tov, 2021; Jang et al., 2024; Spengler, 2023), we focus our exploration on these two types of systems.

Amongst SEs, Google continues to be the most popular SE of all time, followed by Bing (Bianchi, 2024). To elicit responses from Google and Bing, we use their official APIs, i.e., Google Custom Search JSON API[2] and Bing Web Search API[3]. For the Google API, we use all default parameters except that we set the language to English using the key–value pair {`'lr': 'lang_en'`}. For the Bing API, we restrict results to the top 10 and set the language and location to English and the USA, respectively, to align with the default Google API configuration: {`'answerCount': 10, 'setLang': 'en', 'mkt': 'en-US'`}.

In the case of LLM agents, OpenAI's ChatGPT is most commonly known and used by autistic users (Jang et al., 2024), which is why we consider it in our exploration, along with Google's Gemini due to its comparable performance to ChatGPT in benchmark question-answering tasks (Gemini Team Google et al., 2023; Achiam et al., 2023). We elicit the direct answer from ChatGPT and Gemini using the official APIs of `GPT-3.5 Turbo`[4] and `Gemini Pro`[5] models, respectively which are the models driving ChatGPT and Gemini's response generation at the time of our study (OpenAI, 2022; Liedtke and O'Brien, 2023; Milmo, 2023). Like for SE responses, we limit LLM agent responses to English. Note that all API calls are made independently and anonymously to control for the influence of user profiling on the elicited responses.

3.2 Data Collection

Logs from mainstream IASs are generally scarce, especially when capturing the interactions initiated by autistic individuals specifically. Consequently, to enable the analysis of accessibility of IASs when responding to autistic individuals' inquiries, we construct *synthetic logs* that capture the content autistic individuals would be exposed to when interacting with an IAS for information search. For this, we require queries that reflect the common information needs of the target population as a starting point. Considering the different needs and varied responses to stimuli of autistic individuals (CDC, 2024a; WHO, 2023), the queries need to capture the various areas of interest for the target audience. Unfortunately, query samples from existing studies on autistic users' search behaviour are either proprietary data (Yechiam and Yom-Tov, 2021; Alzahrani et al., 2022) or cannot be shared publicly due to regulations protecting the privacy of vulnerable populations (Milton and Pera, 2023), which limits our access to queries formulated by a diverse group of real autistic users.

To overcome this challenge, motivated by similar works involving other vulnerable and under-represented populations, such as individuals afflicted with mental health issues, we adopt a synthetic approach that leverages data from public social media interactions of self-disclosed individuals belonging to a target population to capture their common needs and interests (e.g., De Choudhury et al., 2016; Paul and Dredze, 2011; De Choudhury et al., 2014; Ríssola et al., 2022). Specifically, we follow the method used in (Milton and Pera, 2023) and extract common noun phrases from social media posts created by individuals who report themselves to belong to a target population. Through this strategy, we isolate phrases containing terminology used by autistic individuals on topics of common interest for this population, which can serve as a proxy for queries that reflect the target user

2. `https://developers.google.com/custom-search/v1/overview`
3. `https://learn.microsoft.com/en-us/bing/search-apis/bing-web-search/overview`
4. `https://platform.openai.com/docs/models/gpt-3-5-turbo`
5. `https://cloud.google.com/vertex-ai/generative-ai/docs/model-reference/gemini`

group's common information needs. In our case, noting the popular use of Reddit posts to study the topics, linguistic patterns, and behavioural aspects typical for autistic users (e.g., Bahrke, 2024; Varghese et al., 2024; Thom-Jones et al., 2024; Fong et al., 2025), we turn to posts made by self-reported autistic users from 16 subreddits (Medvedev et al., 2019)[6]. In doing so, we consider data addressing different perspectives within the autistic community, enabling a wide and diverse coverage of topics and vocabulary commonly used by autistic people. See Appendix A for the description of each subreddit that was scraped for data collection.

Using the official Python Reddit API wrapper (PRAW[7]), we collected the title and text body of 11,100 Reddit posts[8]. To transform the posts into synthetic queries, we extracted the common noun phrases using `KeyBERT` (Grootendorst, 2020), a BERT embeddings-based keyphrase extraction technique, which yielded approximately 28,880 key phrases. As stated in (Van Rijsbergen, 1979), it is common for certain terminologies to be used very frequently in any given corpus to a point where the terms do not capture any distinct information and instead serve as corpus-specific stopwords. To eliminate such terms from our collection of keyphrases, we turn to Zipf's law (Zipf, 2016), which states that the frequency of any term is inversely proportional to its rank in a frequency table. Specifically, following the stopword-removal method used in (Sahu et al., 2023; Lo et al., 2005; Courseault Trumbach and Payne, 2007), we rank the keyphrases in a descending order of their frequency of occurrence (grouped by their n-gram length). We plot the keyphrases in the order of their rank and estimate a maximum threshold of occurrence for each group (listed in Table 1). Keyphrases with a frequency above the threshold are considered as stopwords and therefore excluded, leading to a collection of 7000 key phrases as candidate synthetic queries.

n-gram	**Max # of posts an included key phrase can occur in**
1	400
2	100
3	10
4	4

Table 1: Upper limit of occurrence for key phrases of different n-gram lengths to exclude in-context stopwords.

We use a sample of the most common key phrases for each n-gram length as the set of queries that capture the information needs of autistic users. Given the impact of query lengths in biasing search results (Alaofi et al., 2022; Pera et al., 2023), we pay particular attention to ensure that the set of synthetic queries follows a realistic query length distribution. However, due to the lack of information on such a distribution specifically for autistic

6. Note that no personally identifiable information was collected, and all content was de-identified before use. All collected data was discarded after the completion of our experiments.
7. `https://github.com/praw-dev/praw`
8. The data was extracted from Reddit in April 2024 and comprises posts created by self-reported autistic individuals between April 2019 and April 2024.

users at the time of the study, we turn to the Yahoo! Search Query Tiny samples dataset[9] (Yahoo!, 2009; Blanco et al., 2011), as it comprises queries formulated by real people and therefore would represent a query length distribution realistic at least for a generic user[10]. Specifically, we compute the proportion of queries for each n-gram length and mimic those proportions to generate set $\mathtt{Q_A}$ comprising 250 queries of which 78 are unigram queries, 78 bigram queries, 70 trigram queries, and the remaining 24 of n-gram length ≥ 4. The queries in $\mathtt{Q_A}$ range from phrases directly related to ASD, such as "autism", "autism diagnosis", and "autistic burnout", to other mental health related phrases like "crippling social anxiety", "sensory overload", and "executive dysfunction", as well as more generic topics like "part time job", "high school", and "facial expression".

Different people express the same information need in different ways, which in turn has a strong influence on the response produced by an IAS (Alaofi et al., 2022; Pera et al., 2023). Given that diagnosed autistic users tend to report their diagnosis in their search query, we create a second *variation* of the synthetic queries in $\mathtt{Q_A}$ to reflect the information needs of autistic individuals covered in $\mathtt{Q_A}$ while also informing the system of the user's condition. To reflect autistic individuals' natural style of reporting their condition in our synthetic queries, we refer to the phrases used by autistic individuals to report their condition in their search query, as identified by Yechiam and Yom-Tov (2021) from the query log the authors used to study the online search behaviour of autistic individuals. The query log used in the study is proprietary; hence, we rely on the examples reported in the manuscript, such as *"I'm autistic"*, *"I'm on the autism"* and *"I am on the autism"*. Particularly, we create two sets $\mathtt{Q'_A}$ and $\mathtt{P'_A}$ to probe SE and LLM agent responses, respectively, with each set representing a distinct variation of queries in $\mathtt{Q_A}$. In the case of $\mathtt{Q'_A}$, we simply append the phrase "*I'm autistic*" at the end of each query in $\mathtt{Q_A}$; whereas for $\mathtt{P'_A}$, we use a longer approach to mimic the conversational nature of information-seeking in LLM agents and rephrase each query in $\mathtt{Q_A}$ as "I am autistic please explain" + query + "to me". See Table 2 for examples of queries in $\mathtt{Q_A}$, $\mathtt{Q'_A}$, and $\mathtt{P'_A}$.

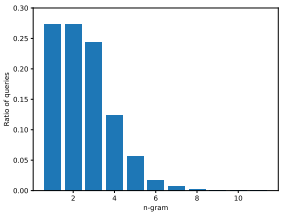
$\mathtt{Q_A}$	$\mathtt{Q'_A}$	$\mathtt{P'_A}$
autism diagnosis	autism diagnosis I'm autistic	I am autistic please explain autism diagnosis to me
common autistic trait	common autistic trait I'm autistic	I am autistic please explain common autistic trait to me

Table 2: Examples of queries in $\mathtt{Q_A}$, $\mathtt{Q'_A}$, and $\mathtt{P'_A}$.

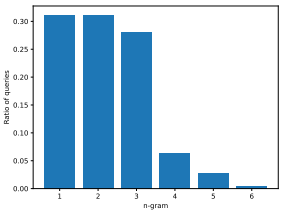
To contextualise whether the accessibility of IAS responses to autistic individuals' (synthetic) queries is inherent to a system's algorithm or shows variances when catering to the information needs of different user groups, we create another query set, $\mathtt{Q_C}$ to capture the topics of interest, vocabulary, and phrasing common amongst the general population. $\mathtt{Q_C}$ consists of 250 queries randomly sampled from the Yahoo! Search Query Tiny sample dataset (Yahoo!, 2009) with a similar n-gram length distribution as $\mathtt{Q_A}$, which we illustrate in Figure 1 along with the n-gram distribution of the original Yahoo! Search Query Tiny sample dataset.

9. The dataset used in this study is no longer publicly available; however, archived versions can be accessed at `https://web.archive.org/web/20250517143901/https://webscope.sandbox.yahoo.com/catalog.php?datatype=l`.

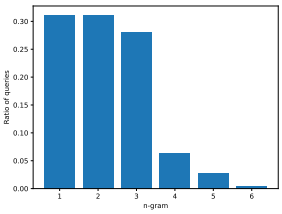
10. Specifically, the dataset comprises a random sample of 4496 queries posted to Yahoo's US SE in January 2009.



(a) Yahoo! Search Query Tiny sample dataset (Yahoo!, 2009).



(b) $\mathtt{Q_C}$



(c) $\mathtt{Q_A}$

Figure 1: n-gram distribution of queries in different query collections.

Using each of the aforementioned query sets, we generate synthetic logs capturing how a given IAS responds to online inquiries. Each log consists of `<query, response>` pairs, where `query` is a query from a given query set, and `response` is the response generated by an IAS for the `query`.

To probe the accessibility of SE responses, we first focus on the Search Engine Result Page (`SERP`) as it is the initial response a searcher receives from an SE for their inquiry. The `SERP response` comprises of the top 10 results retrieved by the SE[11]; each result represented by the associated web title and snippet[12]. Additionally, we examine SE responses focusing on individual retrieved results (`RR`) to investigate the content an autistic user would be exposed to upon clicking on a result from the SERP. The `RR response` consists of the entire text body of the web resources retrieved as the top 10 results by an SE for a `query`. In the case of LLM agents, `response` refers to the direct answer (`ANS`) that an LLM generates for a `query`.

In the remainder of the manuscript, we refer to a synthetic log as a possible combination of three components: **{$\mathtt{Q_A}$, $\mathtt{Q_C}$, $\mathtt{Q'_A}$, $\mathtt{P'_A}$}-{Google (G), Bing (B), ChatGPT (GPT), Gemini (GEM)}-{SERP, RR, ANS}**[13]. For instance, $\mathtt{Q_A}$-G-SERP is the log capturing Google's SERP snippets for each query in $\mathtt{Q_A}$.

3.3 Accessibility Indicators

To examine IAS responses through a web accessibility lens, we rely on 11 quantitative measures, which we refer to as *indicators*, based on WCAG for autistic individuals related to three text-based perspectives that may influence autistic individuals' information seeking: (1) Structure, (2) Readability, and (3) Concreteness. See Table 3 for detailed indicator definitions.

Structure. Some autistic people struggle with reading huge blocks of text, often leading to information overload in them, which could incite challenging behaviours or meltdowns

11. We only consider the top 10 results, i.e., the first SERP, as most users never go past the first result page when searching for information (Sharma et al., 2019).
12. At the time of the study, AI Overviews are not generated consistently and are therefore not included in the collected SERP responses.
13. Note that `SERP` and `RR` are only for SE responses, and `ANS` is only for LLM agent responses.

in autistic individuals, thereby hindering them from engaging with the content (White and Abou-Zahra, 2024; Happé and Frith, 2006). Accessibility guidelines, therefore, suggest presenting textual content as succinctly as possible (Britto and Pizzolato, 2016; Seeman and Cooper, 2024). Given that paragraphs usually are the longest component of a text, we consider the average number of words per paragraph in T (**Average Paragraph Length**) as a measure of *text succinctness*. Moreover, as autistic individuals are known to prefer shorter sentences over long-winded sentences (Yaneva and Evans, 2015), we also examine the text succinctness of a text sample at the sentence-level. Hence, we measure the number of sentences (**Sentence Count**) and the average number of words per sentence (**Average Sentence Length**) as two other accessibility indicators. Further, markup features like lists and headings can improve reading flow for autistic individuals (Britto and Pizzolato, 2016). Therefore, we inspect the structure of T also by the proportion of sentences in T that constitute heading titles (**Ratio of Headings**), list items (**Ratio of List Items**), or a paragraph (**Ratio of Paragraphs**) to assess the *reading flow* of T. Note that we compute the Ratio of Paragraphs for T as the proportion of sentences that are neither headings nor list items. Therefore, the three ratios computed for T add up to 1.

Readability. For an individual to extract information from a resource, they must be able to comprehend the content presented in the resource. For some autistic individuals, texts written in simple and plain language are more suitable for comprehension than texts written in complex sentences (Yaneva et al., 2015; Seeman and Cooper, 2024). Therefore, as our second perspective, we examine the language complexity of T. For this, we turn to two popular readability formulas, which look at different linguistic aspects of a text to determine the reading ease of the text. First, the Flesch Reading Ease Score (**FRES**) (Kincaid et al., 1975) due to its popularity in research, especially in studies pertaining to autistic individuals (Yaneva and Evans, 2015; Yaneva et al., 2015). The FRES score is computed based on the average words per sentence and average syllables per word in a text, and a higher score indicates higher ease of reading. The other readability formula we consider is the Coleman-Liau Readability Index (**CLI**) (Coleman and Liau, 1975) due to its suitability for digital texts (Allen et al., 2022). The index value is computed based on the average number of letters per word instead of syllables, and unlike FRES, a higher CLI index indicates that the text is *less* easy to read.

Concreteness. Autistic people prefer literal language and concrete terminology over figurative and abstract language (Kalandadze et al., 2018; Vulchanova and Vulchanov, 2022). We posit that for a text to be accessible to an autistic person it must be as concrete as possible. To assess the concreteness of T, we estimate the average concreteness of T (**Average Concreteness**) by adopting a lexical approach based on a dataset created by Brysbaert et al. (2014). The dataset consists of concreteness ratings for 40 thousand generally known English lemmas (ranging between 1: maximally abstract and 5: maximally concrete), estimated based on the average of ratings provided by over 4000 participants. Note that, as per this dataset, not all English words include a concreteness score; therefore, we compute the average concreteness of T only based on the words in T that have a concreteness rating in the dataset, i.e, words in T which exactly match those in the dataset. Since the average concreteness does not account for unrated words in T, we also measure the proportion of words in T classified as concrete (**Ratio of Concrete Words**) and those classified as ab-

Accessibility Perspective	Indicator Name	Indicator Formula
Structure	Average Paragraph Length	$ParaLen(T) = \frac{\Sigma_{i=1}^{\lvert P_T \rvert} len(p_i)}{\lvert P_T \rvert}$, $p_i \in P_T$
	Sentence Count	$SC(T) = \lvert S_T \rvert$
	Average Sentence Length	$SentLen(T) = \frac{\Sigma_{i=1}^{\lvert S_T \rvert} len(s_i)}{\lvert S_T \rvert}$, $s_i \in S_T$
	Ratio of Headings	$HeadingSents(T) = \frac{\lvert H_T \rvert}{\lvert S_T \rvert}$
	Ratio of List Items	$ListSents(T) = \frac{\lvert L_T \rvert}{\lvert S_T \rvert}$
	Ratio of Paragraphs	$ParaSents(T) = 1 - (\frac{\lvert H_T \rvert}{\lvert S_T \rvert} + \frac{\lvert L_T \rvert}{\lvert S_T \rvert})$
Readability	FRES	$\text{FRES}(T) = 206.835 - 1.015 * \frac{\lvert W_T \rvert}{\lvert S_T \rvert} - 84.6 * \frac{syllables(T)}{\lvert W_T \rvert}$
	CLI	$\text{CLI}(T) = 0.0588 * \frac{letters(T) * 100}{\lvert W_T \rvert} - 0.296 * \frac{\lvert S_T \rvert * 100}{\lvert W_T \rvert} - 15.8$
Concreteness	Average Concreteness	$Conc(T) = \frac{\Sigma_{i=1}^{\lvert W_T \rvert} conc(w_i)}{\Sigma_{i=1}^{\lvert W_T \rvert} e(w_i)}$, $w_i \in W_T$; $e(w_i) = \begin{cases} 1, & \text{if } conc(w_i) > 0 \\ 0, & \text{otherwise} \end{cases}$
	Ratio of Concrete Words	$ConcreteWords(T) = \frac{\Sigma_{i=1}^{\lvert W_T \rvert} c(w_i)}{\lvert W_T \rvert}$, $c(w_i) = \begin{cases} 1, & \text{if } conc(w_i) \geq 4 \\ 0, & \text{otherwise} \end{cases}$
	Ratio of Abstract Words	$AbstractWords(T) = \frac{\Sigma_{i=1}^{\lvert W_T \rvert} a(w_i)}{\lvert W_T \rvert}$, $a(w_i) = \begin{cases} 1, & \text{if } 0 < conc(w_i) < 4 \\ 0, & \text{otherwise} \end{cases}$

Table 3: Definitions of indicators used in our study to assess the accessibility of a text sample T, where $len(\cdot)$ yields the number of words, $syllables(\cdot)$ the number of syllables, $letters(\cdot)$ the number of letters, $conc(\cdot)$ the concreteness rating estimated using (Brysbaert et al., 2014), and $\lvert \cdot \rvert$ the size of a set; the variable W_T is the set of words in T, P_T the set of paragraphs, S_T the set of sentences in T. H_T is the set of sentences in S_T that are headings, and L_T is the set of sentences in S_T that are list items. Note that a sentence in T is either a heading, list item, or part of a paragraph, and therefore, the ratios of headings, list items, and paragraphs add up to 1.

stract (**Ratio of Abstract Words**) to contextualise the average concreteness estimation. The classification is done based on the categorisation rule used in the original study, i.e., words rated between 1-3 are abstract and those rated between 4-5 are concrete (Brysbaert et al., 2014).

3.4 Accessibility Profile

To get an overview of a text sample (e.g. a SERP snippet, full text of a resource retrieved by an SE, and answer generated by an LLM agent) from an accessibility point of view, we turn to other works on online content analysis. Specifically, we are inspired by works which create

a "profile" to capture various aspects of an item, such as the emotional undertone of IAS responses (Kazai et al., 2019; Landoni et al., 2020), implicit emotional and vocabulary-based triggers in SE responses for individuals suffering from mental health disorders (Milton and Pera, 2023), etc. A profile in these works is operationalised as a vector comprising multiple components to represent the different perspectives of analysis. In our case, to examine a sample from the perspectives of Structure, Readability, and Concreteness, we create an *accessibility vector* which accounts for all the indicators described in Section 3.3 and is illustrated in Figure 2. An accessibility vector, however, provides insights into individual samples but not an IAS in general, which is the focus of our study. Hence, we create an overall *accessibility profile* of an IAS, $\mathbf{AP}(\texttt{log})$, by taking an average of text sample vectors of all IAS responses in a synthetic `log` (§3.2). Note that an accessibility profile in our study is not meant to serve as a set of metrics that can determine whether an IAS is accessible for autistic users. Instead, the accessibility profile of an IAS provides an overview of different (text-based) aspects of the system when responding to the information needs of a user group that are known to impact autistic individuals' information seeking.

Considering an SE provides multiple results for a user's query, we create an accessibility profile for an SE `log` by first averaging text sample vectors grouped by the query, and then computing the average of the resultant query-wise vectors. For LLM agents, since a `log` contains only one response per query, we create a profile by taking an average of all the text sample vectors in the `log`. We are also interested in the impact of query length and topic of inquiry on the text accessibility of IAS responses, hence we also generate (i) *n-gram profile*, $\mathbf{AP}_{n-gram}(\texttt{log})$, by averaging the text sample vectors (query-wise vectors in case of SE-based logs) of responses to queries of $n-gram$ length; and (ii) *inquiry topic profile*, $\mathbf{AP}_{topic}(\texttt{log})$, by taking an average of the text sample vectors of responses to queries related to an inquiry topic category.

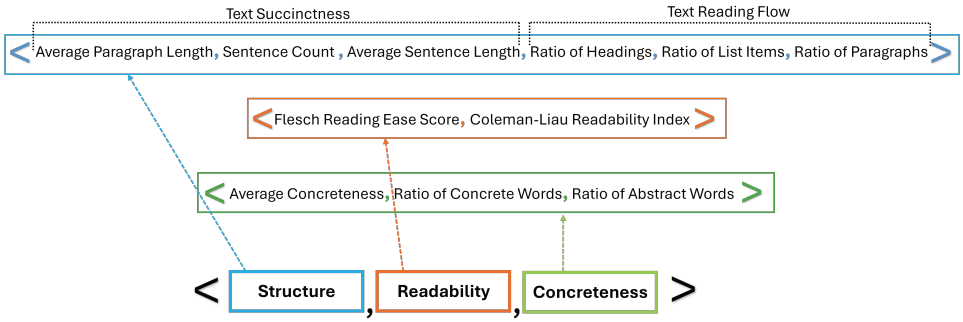


Figure 2: Structure of the accessibility vector representing text accessibility of each text sample.

Example. To illustrate the creation of a profile for a given IAS, consider a query set $\texttt{Q}_\texttt{E}$, comprising queries "banana", "white melon", and "motorcycle". We use the query set to build a synthetic SE based `log`: $\texttt{Q}_\texttt{E}$-G-SERP, consisting of Google's top 10 SERP results (i.e., web title and snippet of top 10 retrieved webpages) per query in $\texttt{Q}_\texttt{E}$. To build

an overall accessibility profile of Google using this `log`, we first average the accessibility vectors of results per query, resulting in 3 query-wise vectors. We then take an average of the 3 vectors to obtain AP($\mathtt{Q_E\text{-}G\text{-}SERP}$). To create a 1-gram profile based on the same log, we compute the average of vectors corresponding to responses for queries "banana" and "motorcycle" respectively, and the 2 resultant query-wise vectors are averaged to create $\text{AP}_{1-gram}(\mathtt{Q_E\text{-}G\text{-}SERP})$. For an inquiry topic profile pertaining to results for queries in the category "fruit", we average the vectors of results for queries "banana" and "white melon" respectively, and then the two vectors are averaged to obtain $\text{AP}_{fruit}(\mathtt{Q_E\text{-}G\text{-}SERP})$.

4 Examining the Accessibility of IAS Responses for Autistic Users

Here, we describe the results of the experiments we conducted to answer **RQ1**.

4.1 Variation across IAS

To assess the overall text accessibility of the considered SEs and LLM agents when responding to autistic individuals' inquiries, we juxtapose the accessibility profiles of all $\mathtt{Q_A}$ based logs reported in Table 4 at the component level. Note that unless stated otherwise, accessibility indicator scores are significantly different (one-way ANOVA $p < 0.05$) across the profiles.

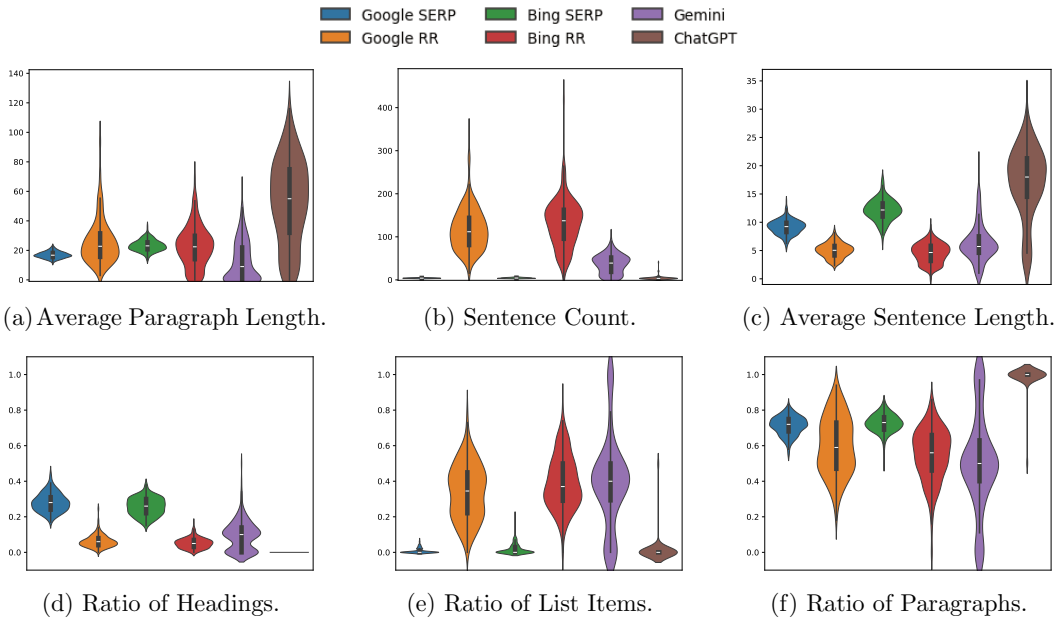
Accessibility Perspective	Accessibility Indicator	AP($\mathtt{Q_A\text{-}G\text{-}SERP}$)	AP($\mathtt{Q_A\text{-}G\text{-}RR}$)	AP($\mathtt{Q_A\text{-}B\text{-}SERP}$)	AP($\mathtt{Q_A\text{-}B\text{-}RR}$)	AP($\mathtt{Q_A\text{-}GEM\text{-}ANS}$)	AP($\mathtt{Q_A\text{-}GPT\text{-}ANS}$)
Structure	ParaLen ↓	16.86	26.11	23.18	24.8	**12.49**	52.87
	SC ↓	**4.08**	117.61	4.43	133.16	37.01	4.17
	SentLen ↓	9.14	5.03	12.16	**4.52**	6.27	17.06
	HeadingSents ↑	**0.28**	0.06	0.26	0.05	0.09	0
	ListSents ↑	0.01	0.34	0.02	0.39	**0.4**	0.02
	ParaSents ↓	0.71	0.6	0.73	0.55	**0.51**	0.98
Readability	FRES ↑	**61.25**	58.78	57.43	59.01	42.06	47.03
	CLI ↓	**10.83**	11.89	11.74	11.79	13.83	12.13
Concreteness	Conc ↑	2.41	**2.43**	2.36	2.42	2.35	2.39
	ConcreteWords ↑	0.04	0.04	0.04	0.04	0.05	**0.06**
	AbstractWords ↓	0.43	0.39	0.54	**0.37**	0.48	0.68

Table 4: Accessibility Profiles of Synthetic Logs based on $\mathtt{Q_A}$. Each indicator score is significantly different (ANOVA one-way, $p < 0.05$) across logs. The best score of each accessibility indicator is **bolded**, where ↓ next to an indicator implies lower values to be better, and ↑ implies higher values to be better.

Structure. Comparing the components pertaining to text succinctness in the profiles from Table 4 (illustrated in Figures 3a-3c), we find that AP($\mathtt{Q_A\text{-}GEM\text{-}ANS}$) has the lowest average paragraph length, and AP($\mathtt{Q_A\text{-}GPT\text{-}ANS}$) the highest. Amongst the logs obtained from SEs, AP($\mathtt{Q_A\text{-}G\text{-}SERP}$) has notably shorter average paragraph length compared to AP($\mathtt{Q_A\text{-}G\text{-}RR}$), while the score is similar for AP($\mathtt{Q_A\text{-}B\text{-}SERP}$) and AP($\mathtt{Q_A\text{-}B\text{-}RR}$). On a sentence level, AP($\mathtt{Q_A\text{-}G\text{-}RR}$) and AP($\mathtt{Q_A\text{-}B\text{-}RR}$) have the highest sentence count but the shortest average sentence length. In fact, we observe an inverse relation between the sentence count and average sentence length across all profiles, which suggests a trade-off between the number

of sentences and their length on average, perhaps to maintain a balance in overall text succinctness across all IAS responses.

In the case of components related to reading flow (Figures 3d-3f), the ratio of paragraphs is higher than the ratios of headings and list items across all profiles, particularly in AP(Q$_A$-GPT-ANS). In AP(Q$_A$-G-SERP) and AP(Q$_A$-B-SERP), the sentences are mostly divided as paragraphs and then headings; in AP(Q$_A$-G-RR), AP(Q$_A$-B-RR), and AP(Q$_A$-GEM-ANS), sentences mostly constitute a paragraph followed by list items. This implies that compared to other IAS, ChatGPT responses on average comprise longer blocks of text, which could impede autistic individuals' reading flow. Regarding the structure of responses produced by the other IAS, the distribution of sentences across markup features is distinct in each profile, implying that each IAS uses a different strategy to facilitate reading flow in its produced responses.



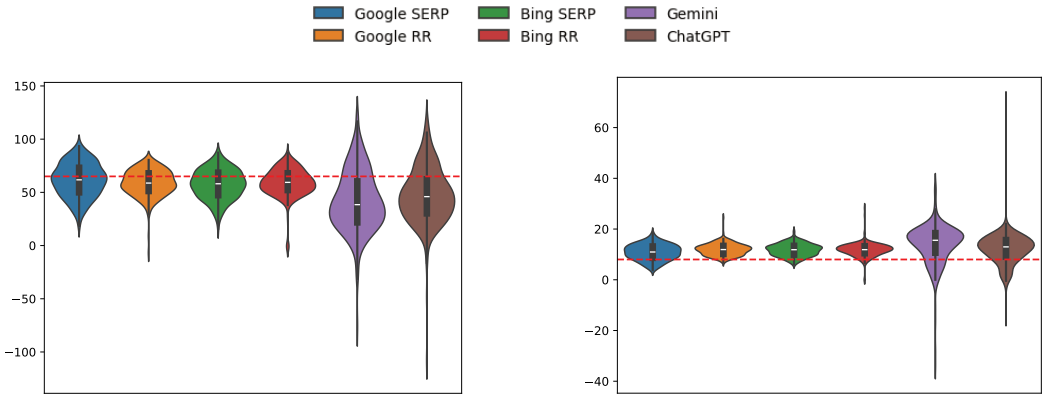
(a) Average Paragraph Length. (b) Sentence Count. (c) Average Sentence Length.

(d) Ratio of Headings. (e) Ratio of List Items. (f) Ratio of Paragraphs.

Figure 3: Text structure scores for logs generated using Q$_A$.

Readability. Yaneva et al. (2015) suggest that for a text to be readable by an autistic person, it must have a FRES score ≥ 65, which is text suitable for a student at grade 8 or 9 in the US school system (Flesch, 2016). In the case of CLI, the integer closest to the computed CLI score indicates the minimum US school grade a reader must have graduated from to be able to read the scored text, so for a text to be readable by an 8^{th} or 9^{th} grade student, the text should have a CLI score ≤ 8. When considering the FRES and CLI scores of the profiles from Table 3, none of the profiles meet either of the readability criteria to be deemed suitable for autistic individuals. From Figures 4a and 4b illustrating the distribution of readability scores of IAS responses within each log, we also note that most responses do not meet the readability criteria. We do however observe notable differences across the profiles with AP(Q$_A$-G-SERP) having the highest FRES score and lowest CLI score,

whereas AP($\texttt{Q}_\texttt{A}$-GEM-ANS) has the lowest FRES score and highest CLI score. This implies that while none of the IASs produce responses that would be ideal for autistic individuals' reading ability, snippets in Google SERP, in general, are the easiest to read and Gemini answers the most challenging.



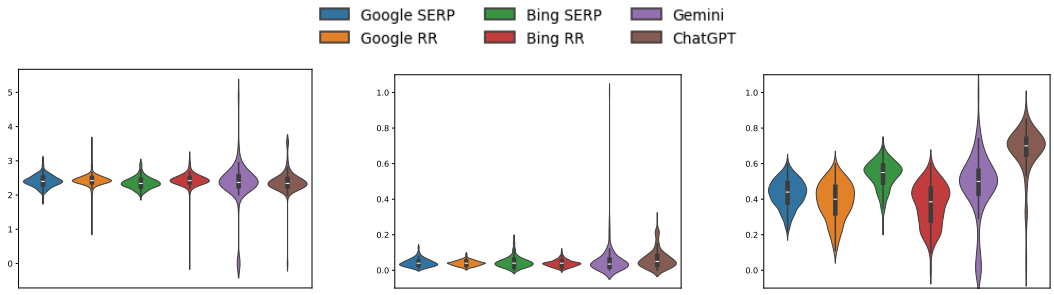
(a) FRES. (b) CLI.

Figure 4: Text readability scores for logs generated using $\texttt{Q}_\texttt{A}$. The dotted red line illustrates the minimum FRES score in (a) and maximum CLI score in (b) an IAS response must have to be considered readable for an autistic person.

Concreteness. Comparing the profiles in Table 4 from the concreteness perspective does not highlight any notable differences. All profiles have an average concreteness score ranging between 2.35 for AP($\texttt{Q}_\texttt{A}$-GEM-ANS) and 2.43 for AP($\texttt{Q}_\texttt{A}$-G-RR) (Figure 5a). Moreover, from the scores of each profile for the ratio of concrete and abstract words respectively, we observe that all profiles have a higher ratio of abstract words than concrete words, conveying that IAS responses to autistic users' inquiries are, on average, more abstract than concrete in nature.

Examining the ratio of concrete and abstract words (Figures 5b and 5c) collectively in each profile, we find that AP($\texttt{Q}_\texttt{A}$-GPT-ANS) has the highest ratio of words with a concreteness rating followed by AP($\texttt{Q}_\texttt{A}$-B-SERP) and AP($\texttt{Q}_\texttt{A}$-GEM-ANS); whereas AP($\texttt{Q}_\texttt{A}$-G-RR) and AP($\texttt{Q}_\texttt{A}$-B-RR) have the lowest ratio of words with a concreteness rating. Provided the lexicon we use to compute the Concreteness indicator scores consists of over 40000 commonly known English lemmas (Brysbaert et al., 2014), we infer that the LLM agent responses consist of more commonly known English words on average compared to SE responses, except for Bing SERP snippets. Considering that autistic individuals find it easier to comprehend a text written using vocabulary they are familiar with (Britto and Pizzolato, 2016; WebAIM, 2025), we infer that although resources retrieved by Google and Bing, in general, have a higher average concreteness than LLM agent responses, autistic individuals would find it easier to comprehend the familiar words in the answers generated by the agents compared to the unfamiliar words in the SE retrieved resources. This implies that autistic individuals' may perceive the information provided in an LLM agent to be more tangible compared

to the information provided in resources retrieved by SE and the corresponding snippets presented in their SERP.



(a) Average Concreteness. (b) Ratio of Concrete Words. (c) Ratio of Abstract Words.

Figure 5: Text concreteness scores for logs generated using $\mathtt{Q_A}$.

4.2 Impact across User Groups

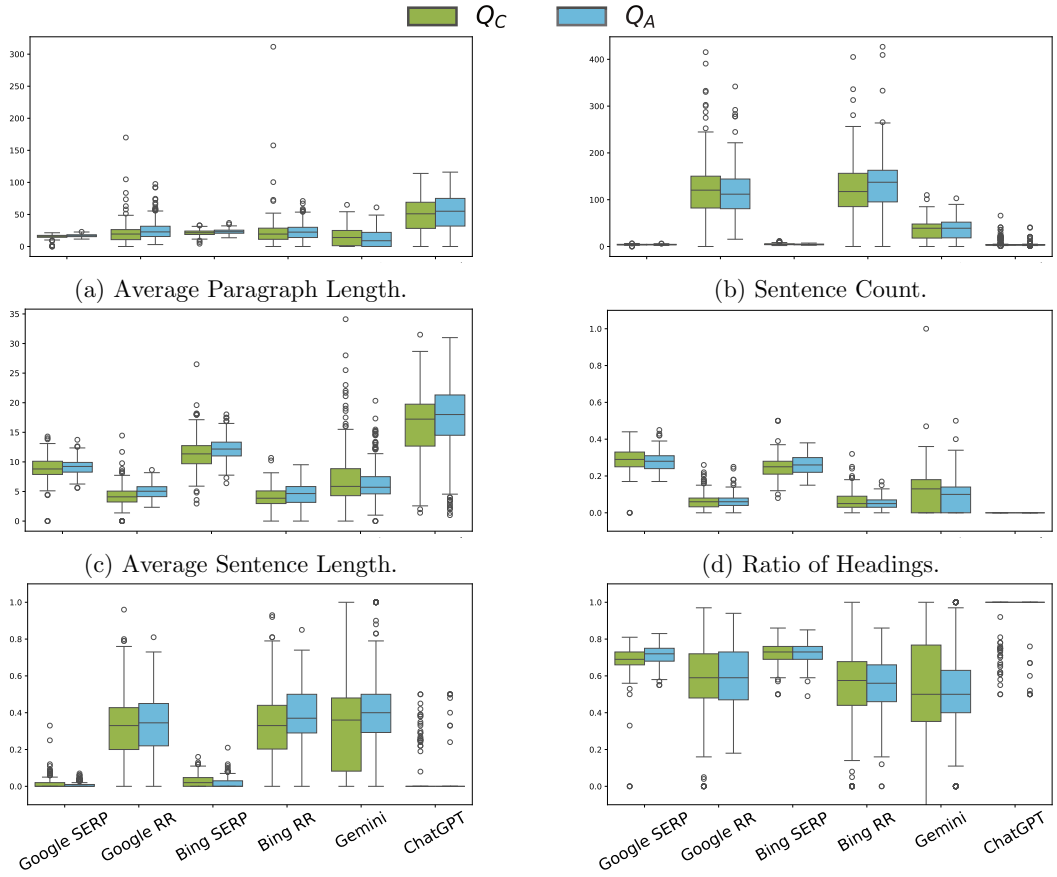
To contextualise whether text accessibility-based aspects of IAS responses studied in this work are consistent regardless of which user group the information need originates from, we compare the accessibility profile generated for a $\mathtt{Q_A}$-based log with the profile generated for its corresponding $\mathtt{Q_C}$-based log (reported in Table 5 and illustrated in Figures 6-8), using an independent T-test ($p < 0.05$) for significance testing.

IAS	Synthetic Log	Structure					Readability			Conceteness		
		Para Len ↓	SC ↓	Sent Len ↓	Heading Sents ↑	List Sents ↑	Para Sents ↓	FRES ↑	CLI ↓	Conc ↑	Concrete Words ↑	Abstract Words ↓
Google SERP	$\mathtt{Q_A}$-G-SERP	16.86	4.08	9.14	0.28	0.01	0.71	61.25	10.83	2.41	0.04	0.43
	$\mathtt{Q_C}$-G-SERP	**15.57 ***	**4**	**8.93**	**0.29**	**0.02 ***	**0.68 ***	**63.01**	**11.03**	**2.43**	**0.05 ***	**0.31 ***
Google RR	$\mathtt{Q_A}$-G-RR	**26.11**	**117.61**	5.03	0.06	**0.34**	0.6	**58.78**	**11.89**	2.43	0.04	0.39
	$\mathtt{Q_C}$-G-RR	26.95	122.48	**4.24 ***	0.06	0.32	**0.58**	56.44	12.58	2.43	0.04	**0.28 ***
Bing SERP	$\mathtt{Q_A}$-B-SERP	23.18	**4.43**	12.16	**0.26**	0.02	0.73	57.43	**11.74**	2.36	0.04	0.54
	$\mathtt{Q_C}$-B-SERP	**21.23 ***	4.9 *	**11.38 ***	0.25	**0.03 ***	**0.72**	**61.55 ***	11.9	**2.5 ***	**0.06 ***	**0.37 ***
Bing RR	$\mathtt{Q_A}$-B-RR	24.8	133.16	4.52	0.05	**0.39**	0.55	**59.01**	11.79	**2.42**	0.04	0.37
	$\mathtt{Q_C}$-B-RR	**21.85**	**120.62 ***	**3.98 ***	**0.07 ***	0.33 *	**0.53**	56.58	**10.98 ***	2.36	0.04	**0.26 ***
Gemini	$\mathtt{Q_A}$-GEM-ANS	**12.49**	**37.01**	**6.27**	0.09	**0.4**	**0.51**	42.06	13.83	2.35	0.05	0.48
	$\mathtt{Q_C}$-GEM-ANS	15.77 *	42.32	7.21 *	**0.12 ***	0.34	0.54	**50.5 ***	**12.49**	**2.41**	0.05	**0.38 ***
ChatGPT	$\mathtt{Q_A}$-GPT-ANS	52.87	**4.17**	17.06	0	0.02	0.98	47.03	12.13	2.39	0.06	0.68
	$\mathtt{Q_C}$-GPT-ANS	**49**	5.45 *	**15.98 ***	0	**0.03**	**0.97**	**55.2 ***	**11.05 ***	**2.49 ***	**0.08 ***	**0.57 ***

Table 5: Accessibility Profiles of Synthetic Logs based on $\mathtt{Q_A}$ and $\mathtt{Q_C}$ respectively. An asterisk (*) on the $\mathtt{Q_C}$ based log denotes a significant difference (independent T-test $p < 0.05$) in indicator score compared to the corresponding $\mathtt{Q_A}$ based log. The better indicator score per compared log pair is **bolded**, where ↓ next to an indicator implies lower values to be better, and ↑ implies higher values to be better.

SE Responses. Looking at the profiles generated for Google SERP based logs produced for autistic individuals and the general population's inquiries in Table 5, we find that AP($\mathtt{Q_C}$-G-SERP) has a significantly lower average paragraph length, sentence count, and average

sentence length than AP($\texttt{Q}_\texttt{A}$-G-SERP). AP($\texttt{Q}_\texttt{C}$-G-SERP) also has a significantly lower ratio of paragraphs and a significantly higher ratio of list items compared to AP($\texttt{Q}_\texttt{A}$-G-SERP). From a Concreteness perspective, AP($\texttt{Q}_\texttt{C}$-G-SERP) has a significantly higher ratio of concrete words and a significantly lower ratio of abstract words. Even the average concreteness of AP($\texttt{Q}_\texttt{C}$-G-SERP) is higher than AP($\texttt{Q}_\texttt{A}$-G-SERP) but the difference is not significant. From these results, we infer that the snippets in a Google SERP produced for inquiries by the general population are, on average, more succinct with an easier reading flow and also more concrete than the snippets produced for autistic individuals' inquiries. In terms of Readability, AP($\texttt{Q}_\texttt{C}$-G-SERP) has a higher FRES score but also a higher CLI score than AP($\texttt{Q}_\texttt{A}$-G-SERP), although the differences don't hold true in practice, from which we infer that snippets produced for autistic individuals and general population inquiries are similarly challenging to read for autistic individuals.



(a) Average Paragraph Length.

(b) Sentence Count.

(c) Average Sentence Length.

(d) Ratio of Headings.

(e) Ratio of List Items.

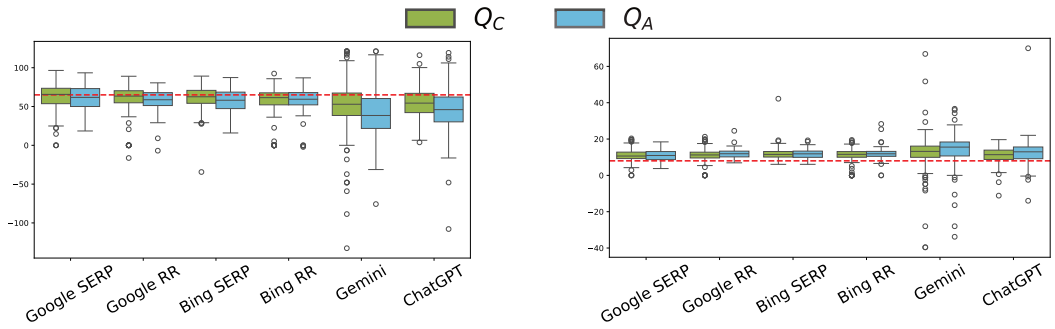
(f) Ratio of Paragraphs.

Figure 6: Text structure scores for logs generated using $\texttt{Q}_\texttt{A}$ and $\texttt{Q}_\texttt{C}$ respectively.

In the case of profiles created for Google RR based logs, AP($\texttt{Q}_\texttt{A}$-G-RR) has a lower average paragraph length and lower number of sentences than AP($\texttt{Q}_\texttt{C}$-G-RR) as well as higher FRES

score and lower CLI score than AP(Q$_C$-G-SERP), but none of the differences are statistically significant. Therefore, we infer that in practice, the resources retrieved by Google in response to inquiries by autistic individuals and the general population are similarly accessible for autistic individuals.

Comparing the accessibility profiles generated for Bing responses reported in Table 5, we find that AP(Q$_C$-B-SERP) has significantly lower average paragraph and sentence lengths, but significantly higher sentence count than AP(Q$_A$-B-SERP). In terms of reading flow, AP(Q$_C$-B-SERP) has a higher ratio of list items and a lower ratio of paragraphs, but also a lower ratio of headings than AP(Q$_A$-B-SERP), although the difference in the ratio of paragraphs does not hold true in practice. From the Readability perspective, AP(Q$_C$-B-SERP) has a significantly higher FRES score but also a higher CLI score than AP(Q$_A$-B-SERP), although the difference in CLI score is not significant. Based on the Concreteness perspective, AP(Q$_C$-B-SERP) has a higher average concreteness, a higher ratio of concrete words, and a lower ratio of abstract words than AP(Q$_A$-B-SERP), with differences across all three components being statistically significant. Based on these results, we infer that Bing SERP snippets are, on average, more succinct with better reading flow, easier to read, and more concrete when produced for the general population's inquiries compared to the snippets produced for autistic individuals' inquiries.



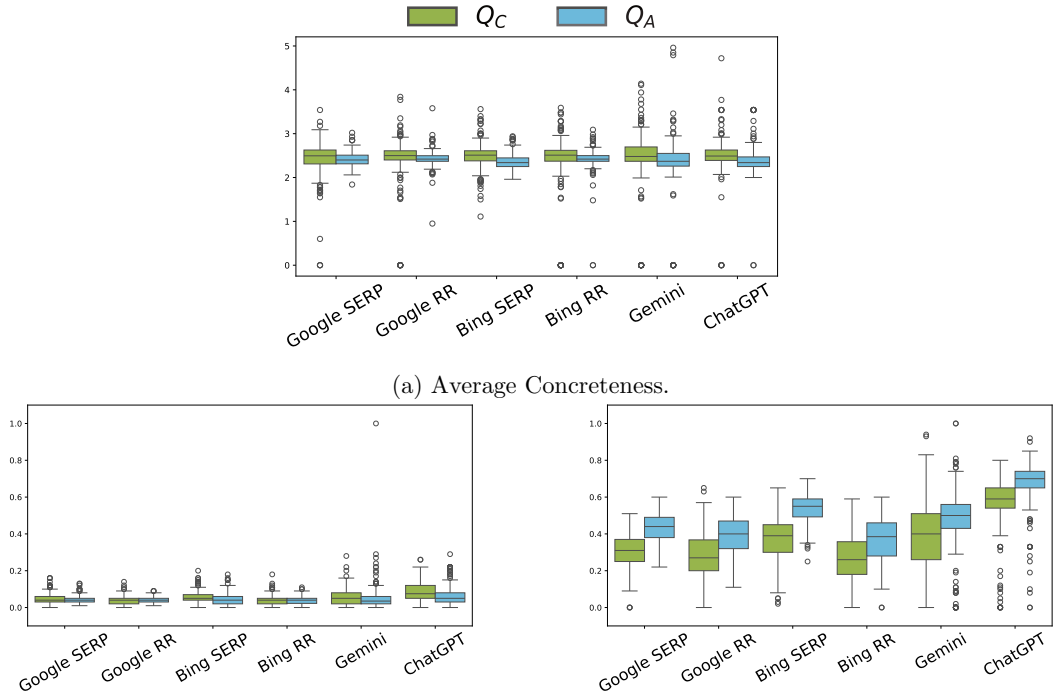
(a) FRES. (b) CLI.

Figure 7: Text readability scores for logs generated using Q$_A$ and Q$_C$ respectively. The dotted red line illustrates the minimum FRES score in (a) and maximum CLI score in (b) an IAS response must have to be considered readable for an autistic person.

The difference in Bing responses becomes even more prominent upon comparing AP(Q$_C$-B-RR) and AP(Q$_A$-B-RR) reported in Table 5, due to the large margin of difference between the two profiles across most component values. AP(Q$_C$-B-RR) has a significantly lower sentence count and a significantly lower average sentence length than AP(Q$_A$-B-RR). In terms of reading flow, AP(Q$_C$-B-RR) has a significantly higher ratio of headings and a significantly lower ratio of paragraphs, but also a significantly lower ratio of list items than AP(Q$_A$-B-RR). From a Readability perspective, AP(Q$_C$-B-RR) has a significantly higher CLI score but a lower FRES score than AP(Q$_A$-B-RR), although the difference in FRES score does not hold true in practice. In terms of Concreteness, AP(Q$_A$-B-RR) has a higher average concreteness than AP(Q$_C$-B-RR) but the difference is not statistically significant; whereas AP(Q$_C$-B-RR) has a

significantly lower ratio of abstract words than AP($\mathtt{Q_A\text{-}B\text{-}RR}$). Based on these results, we find that the resources retrieved by Bing for inquiries initiated by the general population are, in general, more succinct with easier reading flow, easier to read, as well as more concrete for autistic individuals than the resources retrieved for autistic individuals' inquiries.

LLM Agent Responses. Comparing the profiles generated for Gemini responses to autistic individuals and the general population's inquiries, i.e., AP($\mathtt{Q_A\text{-}GEM\text{-}ANS}$) and AP($\mathtt{Q_C\text{-}GEM\text{-}ANS}$) reported in Table 5, we find that AP($\mathtt{Q_C\text{-}GEM\text{-}ANS}$) has higher FRES score and lower CLI score, as well as higher average concreteness and lower ratio of abstract words than AP($\mathtt{Q_A\text{-}GEM\text{-}ANS}$). This implies that from the Readability and Concreteness perspectives, Gemini generates, on average, more accessible responses for the general population's inquiries than for autistic individuals' inquiries. Still, the margin of difference is small and not significant except for the significant difference in the ratio of abstract words. From the Structure perspective, on the other hand, AP($\mathtt{Q_A\text{-}GEM\text{-}ANS}$) has significantly lower average paragraph and sentence lengths and significantly lower sentence count than AP($\mathtt{Q_C\text{-}GEM\text{-}ANS}$). These outcomes indicate that Gemini generates more succinct responses with easier reading flow for autistic users' inquiries compared to the general population's inquiries.



(a) Average Concreteness.

(b) Ratio of Concrete Words.

(c) Ratio of Abstract Words.

Figure 8: Text concreteness scores for logs generated using $\mathtt{Q_A}$ and $\mathtt{Q_C}$ respectively.

In the case of ChatGPT responses, AP($\mathtt{Q_C\text{-}GPT\text{-}ANS}$) has significantly lower average paragraph and sentence lengths than AP($\mathtt{Q_A\text{-}GPT\text{-}ANS}$). Even in terms of reading flow, AP($\mathtt{Q_C\text{-}GPT\text{-}ANS}$) has a significantly lower ratio of paragraphs and a significantly higher ratio of list

items. These results imply that responses in $\mathtt{Q_C}$-GPT-ANS on average are more succinct and have an easier reading flow than the responses in $\mathtt{Q_A}$-GPT-ANS. From the Readability perspective, AP($\mathtt{Q_C}$-GPT-ANS) has a significantly higher FRES score and significantly lower CLI score than AP($\mathtt{Q_A}$-GPT-ANS), implying that ChatGPT generates easier to read responses for general population's inquiries compared to the responses generated for autistic individuals' inquiries. We observe a similar trend from the Concreteness perspective as well, with AP($\mathtt{Q_C}$-GPT-ANS) having a significantly higher average concreteness and ratio of concrete words, and a significantly lower ratio of abstract words compared to AP($\mathtt{Q_A}$-GPT-ANS). Putting all the results together, ChatGPT, like the other considered IAS, produces responses that are more aligned with text accessibility guidelines for autistic individuals when an inquiry is initiated by a user from the general population, compared to the responses generated for autistic individuals' inquiries.

4.3 Influence of Query Phrasing

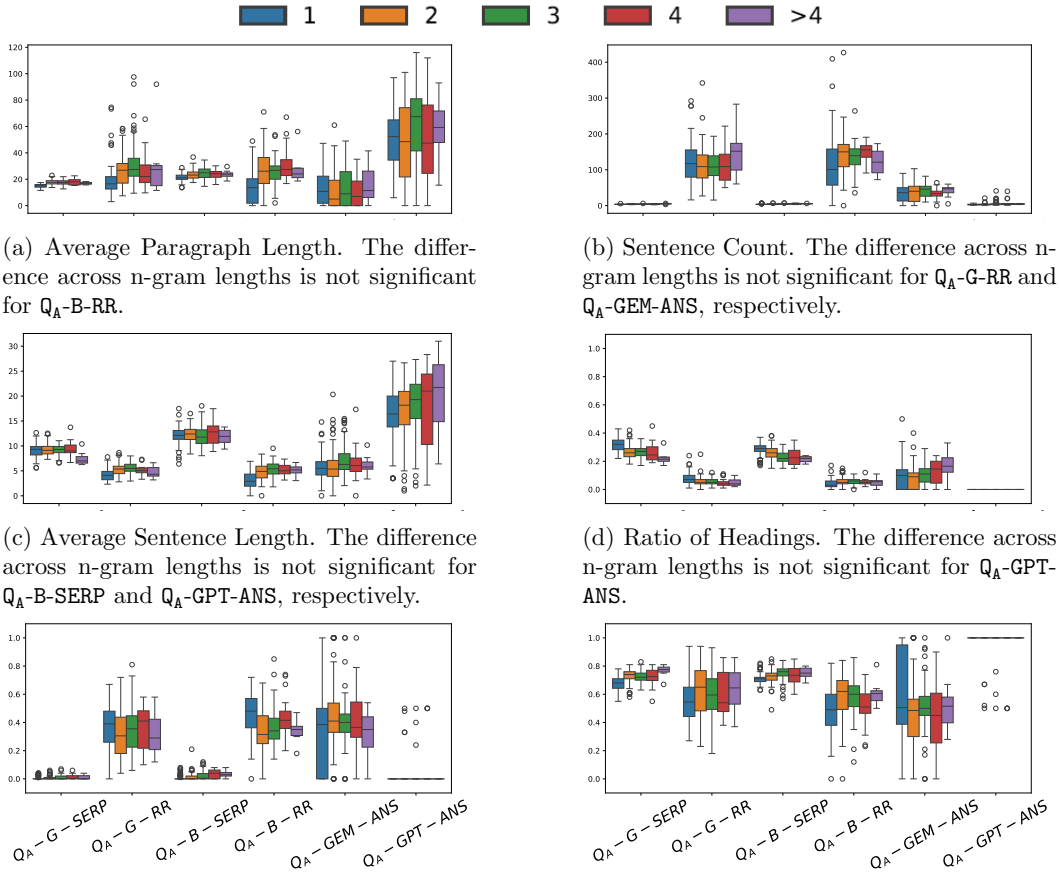
The manner in which a searcher formulates their query to articulate their information need strongly influences the responses produced by an IAS (Alaofi et al., 2022; Pera et al., 2023). Based on this, we explore the potential influence of variation in query phrasing in $\mathtt{Q_A}$ on the text accessibility of elicited IAS responses. Particularly, we examine the impact of variation in (1) the **query length**, i.e., the number of words in the query, and (2) the **inquiry topic**, in our case a binary categorisation indicating whether the autistic users' inquiry pertains to ASD or general topics.

Query Length. We examine the variation in the text accessibility of IAS responses to autistic individuals' inquiries across five categories of query n-gram lengths: unigram, bigram, trigram, 4-gram, and queries longer than 4-gram. For this, we generate $\mathrm{AP}_{n-gram}(\mathtt{log})$ where log is an $\mathtt{Q_A}$ based log as described in Section 3.2 and $n = 1, 2, 3, 4, > 4$. We compare the five profiles generated per log at the component level using one-way ANOVA ($p < 0.05$) to test for statistical significance.

We illustrate in Figure 9 the variation in the text structure scores of IAS responses across queries of different n-gram lengths. For a given IAS, there are statistically significant differences in Structure-related indicator scores across responses for different query lengths. However, we do not observe any prominent trends across or within a log, except for in $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-GPT-ANS})$, where the ratios of headings, list items, and paragraphs remain consistent across the n-gram categories, which implies that the reading flow of ChatGPT responses is unaffected by the query length.

The sentence count in $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-G-SERP})$, $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-B-SERP})$, and $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-GPT-ANS})$, respectively, also remains consistent across the n-gram categories. However, the average sentence and paragraph lengths are consistent only in $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-G-SERP})$ and $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-B-SERP})$ and not in $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-GPT-ANS})$. This implies that the succinctness of Google and Bing SERP responses is unaffected by the query length, but in the case of ChatGPT responses, while the number of sentences remains unaffected on average, the lengths of the sentences and paragraphs vary across query lengths.

Figure 10 illustrates the impact of query length variation on the Readability indicator scores of IAS responses. From the figure, we find that the FRES score decreases significantly and the CLI score increases significantly in $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-GPT-ANS})$ and $\mathrm{AP}_{n-gram}(\mathtt{Q_A}\text{-GEM-}$



(a) Average Paragraph Length. The difference across n-gram lengths is not significant for Q_A-B-RR.

(b) Sentence Count. The difference across n-gram lengths is not significant for Q_A-G-RR and Q_A-GEM-ANS, respectively.

(c) Average Sentence Length. The difference across n-gram lengths is not significant for Q_A-B-SERP and Q_A-GPT-ANS, respectively.

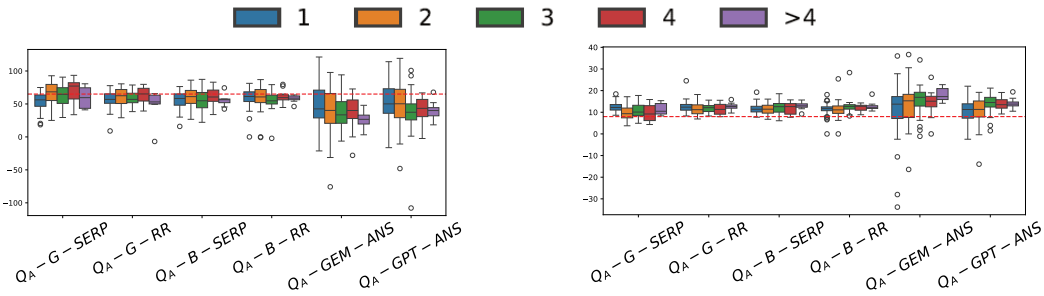
(d) Ratio of Headings. The difference across n-gram lengths is not significant for Q_A-GPT-ANS.

(e) Ratio of List Items. The difference across n-gram lengths is not significant for Q_A-G-SERP and Q_A-G-RR, respectively.

(f) Ratio of Paragraphs.

Figure 9: Text structure scores for Q_A based logs across query n-gram length categories. The difference in indicator scores across n-gram lengths is significant (one-way ANOVA, $p < 0.05$) for a log unless stated otherwise.

ANS) respectively as the query length increases. This implies that for longer queries, both ChatGPT and Gemini generate responses that are harder to read. In case of Bing responses, FRES score in AP_{n-gram}(Q_A-B-SERP) and AP_{n-gram}(Q_A-B-RR) is notably lower and CLI score notably higher when $n > 4$, although only the CLI score difference in AP_{n-gram}(Q_A-B-RR) holds true in practice. For Google responses, the FRES score across AP_{n-gram}(Q_A-G-SERP) and AP_{n-gram}(Q_A-G-RR) is significantly lower when $n = 1$ and $n > 4$. The CLI score in AP_{n-gram}(Q_A-G-SERP) and AP_{n-gram}(Q_A-G-RR) is also prominently higher when $n = 1$ and $n > 4$, however the difference is significant only in AP_{n-gram}(Q_A-G-SERP). These trends suggest that in Bing, longer queries lead to harder-to-read responses, while in Google, extremely short or long queries similarly result in harder-to-read responses.



(a) FRES. The difference across n-gram lengths is not significant for Q_A-B-SERP and Q_A-B-RR.

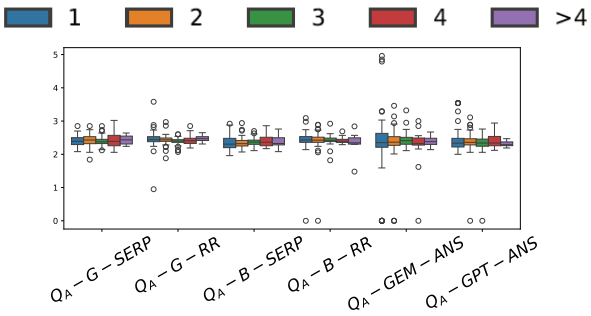
(b) CLI. The difference across n-gram lengths is not significant for Q_A-G-RR and Q_A-B-SERP.

Figure 10: Text readability for Q_A based logs across query n-gram length categories. The difference in indicator scores across n-gram lengths is significant (one way ANOVA, $p < 0.05$) for a log unless stated otherwise. The dotted red line illustrates the minimum FRES score in (a) and maximum CLI score in (b) an IAS response must have to be considered readable for an autistic person.

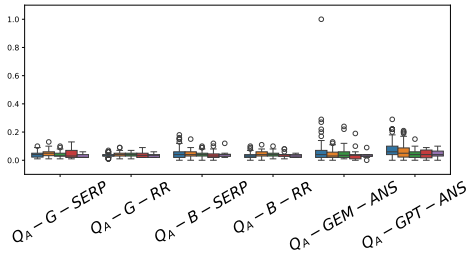
We depict the variation in the concreteness of IAS responses across varying query lengths in Figure 11 and observe notable differences only in the Ratio of Abstract words although the difference is not statistically significant in $\text{AP}_{n-gram}(Q_A\text{-GPT-ANS})$ and $\text{AP}_{n-gram}(Q_A\text{-GEM-ANS})$, respectively. In both $\text{AP}_{n-gram}(Q_A\text{-G-SERP})$ and $\text{AP}_{n-gram}(Q_A\text{-B-SERP})$, the ratio of abstract words is significantly lower when $n \geq 4$ whereas in $\text{AP}_{n-gram}(Q_A\text{-G-RR})$ and $\text{AP}_{n-gram}(Q_A\text{-B-RR})$, the ratio is significantly lower when $n = 1$. This implies that longer queries result in responses that have significantly fewer abstract words in web snippets produced by Google and Bing, but the full content in the corresponding retrieved resource is more abstract in nature than the resources retrieved by the SE for shorter queries.

Inquiry Topic. Considering the impact of ASD in several autistic individuals' quality of life (Lin and Huang, 2019; Mason et al., 2018), inquiry topics related to their condition could be considered a crucial information need for autistic users. Hence, as an initial exploration of the impact of inquiry topic on the accessibility of IAS responses, we investigate if the considered IASs respond differently to autistic users' inquiries related to their condition in comparison to inquiries on more generic topics. For this, we categorise each query in Q_A as an ASD-related inquiry (**D**) or a General inquiry (**G**) based on the constituent keywords in the query. In other words, if a query contains an ASD-related term, it is categorised as an ASD-related inquiry, and a General inquiry otherwise. Due to the lack of a standardised list of ASD-related terms at the time of this study, we rely on public educational resources for autistic people, specifically the glossary terms from ReframingAutism.org[14], a charity run by autistic people in Australia aimed to improve the understanding of Autism through education, resources, and research . The distribution of Q_A queries between the two categories is illustrated in Figure 12. Based on these categories, we produce $\text{AP}_{topic}(\texttt{log})$ for

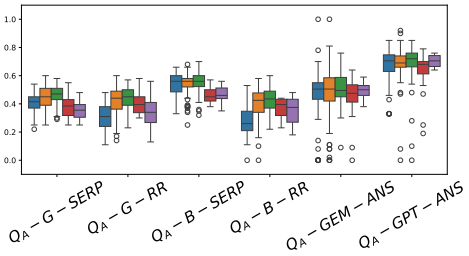
14. The terms were extracted from `https://reframingautism.org.au/service/glossary-terms/` in October 2024.



(a) Average Concreteness. The difference across n-gram lengths is not significant for any logs.



(b) Ratio of Concrete Words. The difference across n-gram lengths is not significant for $\texttt{Q}_\texttt{A}$-B-SERP.



(c) Ratio of Abstract Words. The difference across n-gram lengths is not significant for $\texttt{Q}_\texttt{A}$-GEM-ANS and $\texttt{Q}_\texttt{A}$-GPT-ANS.

Figure 11: Text concreteness scores for $\texttt{Q}_\texttt{A}$ based logs across query n-gram length categories. The difference in indicator scores across n-gram lengths is significant (one way ANOVA, $p < 0.05$) for a log unless stated otherwise.

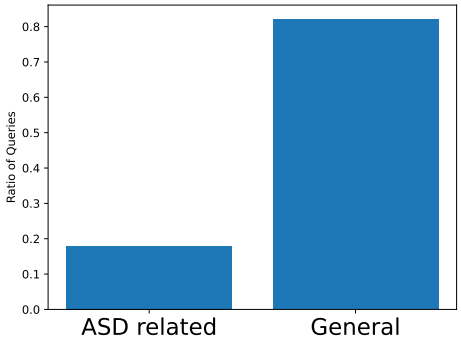


Figure 12: Distribution of $\texttt{Q}_\texttt{A}$ queries between inquiry topic categories.

each $\texttt{Q}_\texttt{A}$ based `log` with *topic* $= \{D, G\}$ and report the generated profiles in Table 6. The

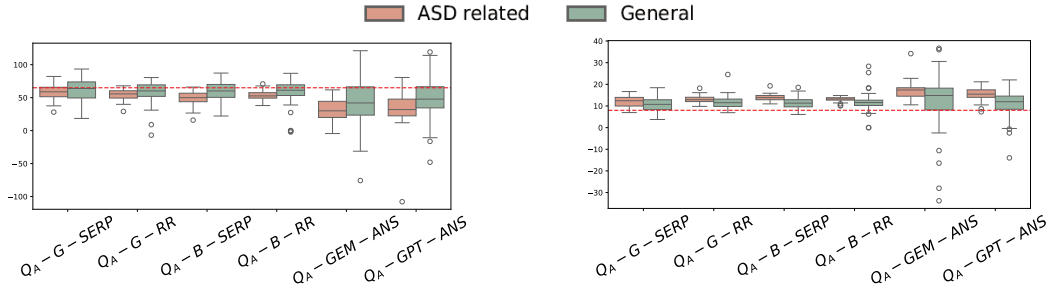
Synthetic Log	Inquiry Topic Category	Structure					Readability			Concreteness		
		Para Len ↓	SC ↓	Sent Len ↓	Heading Sents ↑	List Sents ↑	Para Sents ↓	FRES ↑	CLI ↓	Conc ↑	Concrete Words ↑	Abstract Words ↓
Q_A-G-SERP	ASD	16.71	**3.99**	9.29	0.28	0.01	0.71	58.98	12	2.38	0.03	0.43
	General	**16.89**	4.1	**9.11**	0.28	0.01	0.71	**61.75**	**10.57 ***	**2.41**	**0.04 ***	0.43
Q_A-G-RR	ASD	33.6	**115.97**	5.27	0.06	**0.44**	**0.5**	54.26	13	2.4	0.03	0.41
	General	**24.46 ***	117.97	**4.98**	0.06	0.32 *	0.62 *	**59.78 ***	**11.64 ***	2.44	**0.04 ***	**0.38**
Q_A-B-SERP	ASD	24.45	4.48	12.83	0.25	0.02	0.73	49.48	13.76	2.33	0.04	0.54
	General	**22.9 ***	**4.42**	**12.01 ***	**0.26**	0.02	0.73	**59.18 ***	**11.29 ***	**2.36**	**0.05 ***	0.54
Q_A-B-RR	ASD	41.58	139.8	5.29	**0.06**	**0.46**	**0.48**	53.77	13.13	2.4	0.03	0.4
	General	**21.12 ***	**131.7**	**4.35 ***	0.05 *	0.38 *	0.56 *	**60.17 ***	**11.49 ***	**2.42**	**0.04**	**0.36 ***
Q_A-GEM-ANS	ASD	14.29	38.16	7.43	**0.11**	0.38	0.52	31.68	16.99	2.32	0.04	0.51
	General	**12.09**	**36.76**	**6.02 ***	0.09	**0.4**	**0.5**	**44.33 ***	**13.14 ***	**2.35**	**0.05**	**0.47**
Q_A-GPT-ANS	ASD	**49.96**	5.67	17.24	0	**0.03**	**0.97**	32	16.52	2.26	0.04	**0.67**
	General	53.52	**3.84 ***	**17.03**	0	0.01	0.99	**50.32 ***	**11.17 ***	**2.41 ***	**0.07 ***	0.68

Table 6: $AP_D(\log)$ and $AP_G(\log)$ for all Q_A-based logs. An asterisk (*) on the score for $AP_D(\log)$ denotes a significant difference (independent T-test $p < 0.05$) in the indicator score compared to the corresponding score in $AP_G(\log)$. The better indicator score per compared pair is **bolded**, where ↓ next to an indicator implies lower values to be better, and ↑ implies higher values to be better.

two profiles generated per `log` are compared at the component level using an independent T-test ($p < 0.05$).

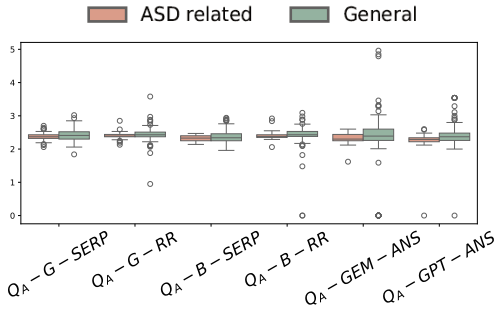
From Table 6, we observe that particularly from the perspective of Readability $AP_G(\texttt{log})$ is closer to autistic people's preferences than the corresponding $AP_D(\texttt{log})$ (Figure 13) with the difference in both FRES and CLI scores statistically significant across all compared profile pairs, except for the difference in FRES score between $AP_G(Q_A\text{-G-SERP})$ and $AP_D(Q_A\text{-G-SERP})$. Even from the perspective of Concreteness, $AP_G(\texttt{log})$ is better suited for autistic individuals than the corresponding $AP_D(\texttt{log})$ (Figure 14), although not all indicator score differences between the compared pairs hold true in practice. The difference in average concreteness is statistically significant only between $AP_D(Q_A\text{-GPT-ANS})$ and $AP_G(Q_A\text{-GPT-ANS})$. The ratio of concrete words is *not* statistically significant between $AP_D(Q_A\text{-B-RR})$ and $AP_G(Q_A\text{-B-RR})$, and between $AP_D(Q_A\text{-GEM-ANS})$ and $AP_G(Q_A\text{-GEM-ANS})$, respectively. The ratio of abstract words is significantly different only between $AP_D(Q_A\text{-B-RR})$ and $AP_G(Q_A\text{-B-RR})$.

From the Structure perspective, we do not observe many significant differences between the compared profile pairs which is also prominent from the similar distributions across most components illustrated in Figure 15. The only exception is when we compare $AP_D(Q_A\text{-B-RR})$ and $AP_G(Q_A\text{-B-RR})$. In terms of text succinctness, $AP_G(Q_A\text{-B-RR})$ has a significantly lower sentence count and significantly lower average paragraph and sentence lengths than $AP_D(Q_A\text{-B-RR})$; whereas based on reading flow, $AP_D(Q_A\text{-B-RR})$ has significantly higher ratios of headings and list items, respectively, along with a significantly lower ratio of paragraphs than $AP_G(Q_A\text{-B-RR})$, implying that when autistic individuals' inquiries are related to ASD, Bing retrieves resources that are structured for easier reading flow, compared to resources retrieved for inquiries on general topics. Overall, we find the topic of inquiry does not influence the structure of IAS responses, except for the reading flow of resources retrieved by Bing.

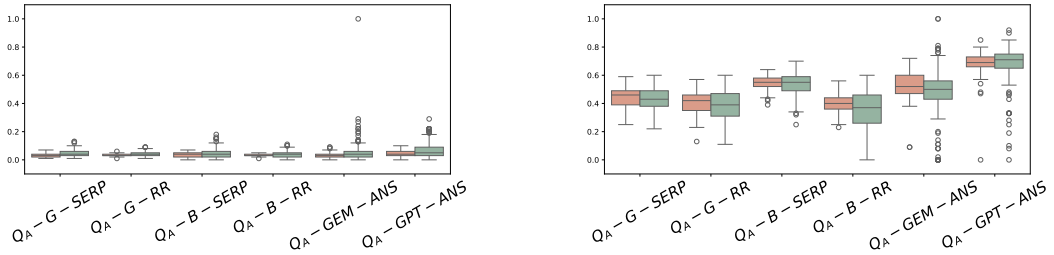


(a) FRES. (b) CLI.

Figure 13: Text readability scores for $\mathtt{Q_A}$ based logs across inquiry topic categories. The dotted red line illustrates the minimum FRES score in (a) and maximum CLI score in (b) an IAS response must be considered readable for an autistic person.



(a) Average Concreteness.

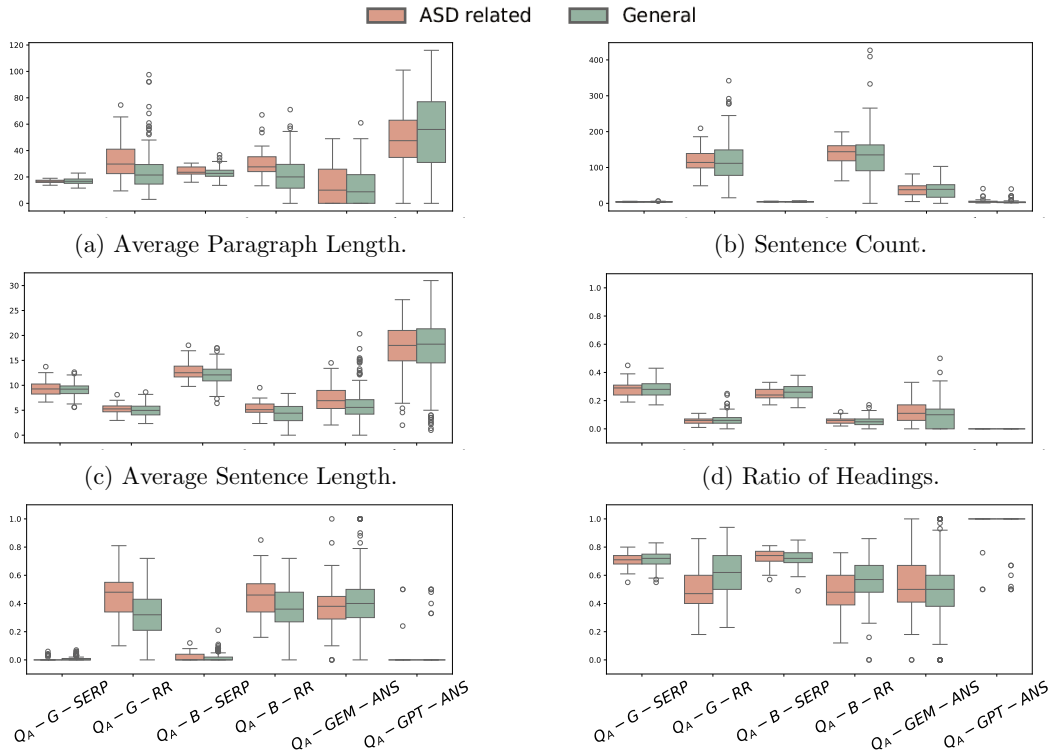


(b) Ratio of Concrete Words. (c) Ratio of Abstract Words.

Figure 14: Text concreteness scores for $\mathtt{Q_A}$ based logs across inquiry topic categories.

4.4 Are the Responses of Mainstream SEs and LLM Agents Accessible to Autistic Users?

To answer RQ1, we scrutinised the Structure, Readability, and Concreteness of the textual content of IAS responses produced for autistic individuals' inquiries. We contextualised whether our observations are due to the systems' general algorithmic behaviour by also assessing the responses produced by the IAS under study for inquiries reflecting the in-



(a) Average Paragraph Length.

(b) Sentence Count.

(c) Average Sentence Length.

(d) Ratio of Headings.

(e) Ratio of List Items.

(f) Ratio of Paragraphs.

Figure 15: Text structure scores for $\mathtt{Q_A}$ based logs across inquiry topic categories.

formation needs of the general population across the same three perspectives. Findings indicate that responses for autistic individuals' queries tend to be more verbose, harder to read, and more abstract in nature compared to responses to inquiries from the general population (§4.1 and §4.2). Further investigations whether the accessibility of an IAS response varies across different types of autistic users' inquiries reveal that responses are significantly harder to read when an autistic user's inquiry is related to ASD than when it is related to other topics (§4.3). Based on these observations, we argue that IAS responses for autistic individuals' inquiries are less closely aligned to the target audience's text accessibility expectations than responses produced for inquiries initiated by the general population.

5 Exploring the Impact of Diagnosis Disclosure in Query Phrasing

In this section, we discuss the results of the experiments we conducted to address **RQ2**.

5.1 Impact on IAS Responses

Knowing that autistic individuals who are aware of their diagnosis often state their condition in their online inquiry (Yechiam and Yom-Tov, 2021) and considering the influence of query

formulation on biasing IAS responses (Alaofi et al., 2022; Pera et al., 2023), we investigate whether rephrasing the original synthetic queries in $\mathtt{Q_A}$ to explicitly state that the user is autistic (as described in Section 3.2) would notably change the response produced by the four IAS we consider in this study.

In the case of Bing and Google, we compare the resources retrieved for inquiries in $\mathtt{Q_A}$ and $\mathtt{Q'_A}$ respectively. For this, we use the Rank Biased Overlap (**RBO**) (Sarica et al., 2022), a measure that computes the similarity between two ranked lists of entities and returns a numeric value between 0 and 1, with 0 indicating that the lists are distinct and 1 indicating that the lists are the same. Specifically, we measure the similarity between SERP produced in response to each query in $\mathtt{Q_A}$ and its counterpart query in $\mathtt{Q'_A}$. Given that an SE may rank different landing pages of the same web resource differently for distinct query variations, we extract the homepage URL of each SERP result and compute the RBO for the two lists of homepages.

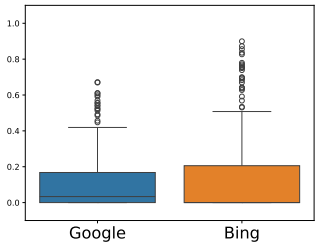
For Gemini and ChatGPT, we compare the direct answers generated by each agent for inquiries in $\mathtt{Q_A}$ and $\mathtt{P'_A}$ respectively. For this, we turn to the Recall-Oriented Understudy for Gisting Evaluation, i.e., ROUGE metrics (Lin, 2004) which measure the recall and precision of a generated summary compared to a (human-made) gold-standard summary, with the score ranging between 0 and 1, where a higher score implies higher similarity between the compared summaries. In our case, we consider the response generated by the LLM agents for queries in $\mathtt{Q_A}$ as the "gold standard" and compare it to the response generated by the agent for the corresponding query in $\mathtt{P'_A}$. Particularly, we measure the similarity between the compared responses—as a whole as well as at the sentence level—using the ROUGE-L metric due to its reported effectiveness on single document summarization tasks (Lin, 2004).

From Figure 16a, we observe that the RBO scores are distributed mostly between 0 and 0.2 for both Google and Bing, which implies that the resources retrieved for the rephrased queries are distinct from the resources retrieved for the original queries. In the case of LLM agent responses, ROUGE-L score at the summary level varies between 0.1 and 0.4 for Gemini and between 0.2 and 0.4 for ChatGPT responses (from Figure 16b), which also depicts low similarity between the LLM agent responses for the rephrased queries when compared to the responses for the original queries. Figure 16c further emphasises the difference at the sentence level, with ROUGE-L scores ranging between 0.1 and 0.2 for Gemini responses and between 0.2 and 0.3 for ChatGPT responses. Based on the similarity metric scores obtained, we infer that informing the IAS that the inquiry is from an autistic user yields notably different responses from all four considered IAS.

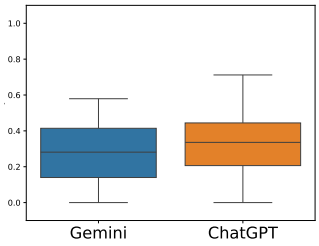
5.2 Influence on Text Accessibility

To investigate whether there is also a difference in the accessibility of the responses produced for the rephrased queries, we generate accessibility profiles for synthetic logs representing SE and LLM agent responses for queries in $\mathtt{Q'_A}$ and $\mathtt{P'_A}$, respectively. For each IAS, we juxtapose its profile, with respect to the counterpart for $\mathtt{Q_A}$ at the component level, as reported in Table 7 and illustrated in Figures 17-19; an independent T-test ($p < 0.05$) for statistical significance.

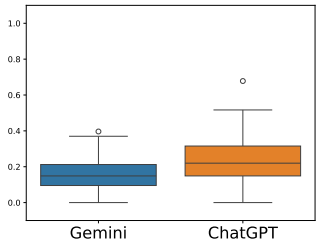
Google. To examine the impact of stating the user's condition in the inquiry on accessibility of Google responses, we compare AP($\mathtt{Q'_A}$-G-SERP) with AP($\mathtt{Q_A}$-G-SERP), and AP($\mathtt{Q'_A}$-G-RR)



(a) RBO score of Google and Bing.



(b) Summary-level ROUGE-L score for Gemini and ChatGPT.



(c) Sentence-level ROUGE-L score for Gemini and ChatGPT.

Figure 16: Similarity scores between IAS responses produced for different query phrasings.

IAS	Synthetic Log	Structure						Readability		Concreteness		
		Para Len ↓	SC ↓	Sent Len ↓	Heading Sents ↑	List Sents ↑	Para Sents ↓	FRES ↑	CLI ↓	Conc ↑	Concrete Words ↑	Abstract Words ↓
Google SERP	Q'_A-G-SERP	19.56	5.05	**8.55**	0.22	0	0.77	**75.56**	**7.2**	**2.54**	0.04	0.5
	Q_A-G-SERP	**16.86** *	**4.08** *	9.14 *	**0.28** *	**0.01** *	**0.71** *	61.25 *	10.83 *	2.41 *	0.04	**0.43** *
Google RR	Q'_A-G-RR	30.98	**96.4**	6.23	0.04	0.24	0.72	**69.08**	**9.75**	2.4	**0.05**	0.52
	Q_A-G-RR	**26.11** *	117.61 *	**5.03** *	**0.06** *	**0.34** *	**0.6** *	58.78 *	11.89 *	**2.43** *	0.04 *	**0.39** *
Bing SERP	Q'_A-B-SERP	24.72	5.12	**11.58**	0.22	**0.03**	0.76	**58.98**	**11.44**	**2.42**	**0.05**	0.55
	Q_A-B-SERP	**23.18** *	**4.43** *	12.16 *	**0.26** *	0.02 *	**0.73** *	57.43	11.74	2.36 *	0.04	**0.54**
Bing RR	Q'_A-B-RR	34.91	139.96	6.12	0.05	0.37	0.57	57.89	11.94	2.41	0.04	0.46
	Q_A-B-RR	**24.80** *	**133.16**	**4.52** *	0.05	**0.39**	**0.55**	**59.01**	**11.79**	**2.42**	0.04 *	**0.37** *
Gemini	P'_A-GEM-ANS	22.19	44.69	7.37	**0.13**	0.37	**0.5**	39.05	15.09	**2.36**	0.04	0.54
	Q_A-GEM-ANS	**12.49** *	**37.01** *	**6.27** *	0.09 *	**0.4**	0.51	**42.06**	**13.83** *	2.35	**0.05**	**0.48** *
ChatGPT	P'_A-GPT-ANS	60.76	7.32	18.51	0	0.02	0.98	45.78	12.66	2.35	0.06	0.7
	Q_A-GPT-ANS	**52.87** *	**4.17** *	**17.06** *	0	0.02	0.98	**47.03**	**12.13**	**2.39**	0.06	**0.68** *

Table 7: Accessibility Profiles of Synthetic Logs based on Q'_A (or P'_A) and Q_A respectively. An asterisk (*) on the log based on Q_A denotes a significant difference (independent T-test $p < 0.05$) in indicator score compared to the corresponding log based on Q'_A or P'_A. The better indicator score per compared log pair is **bolded**, where ↓ next to an indicator implies lower values to be better, and ↑ implies higher values to be better.

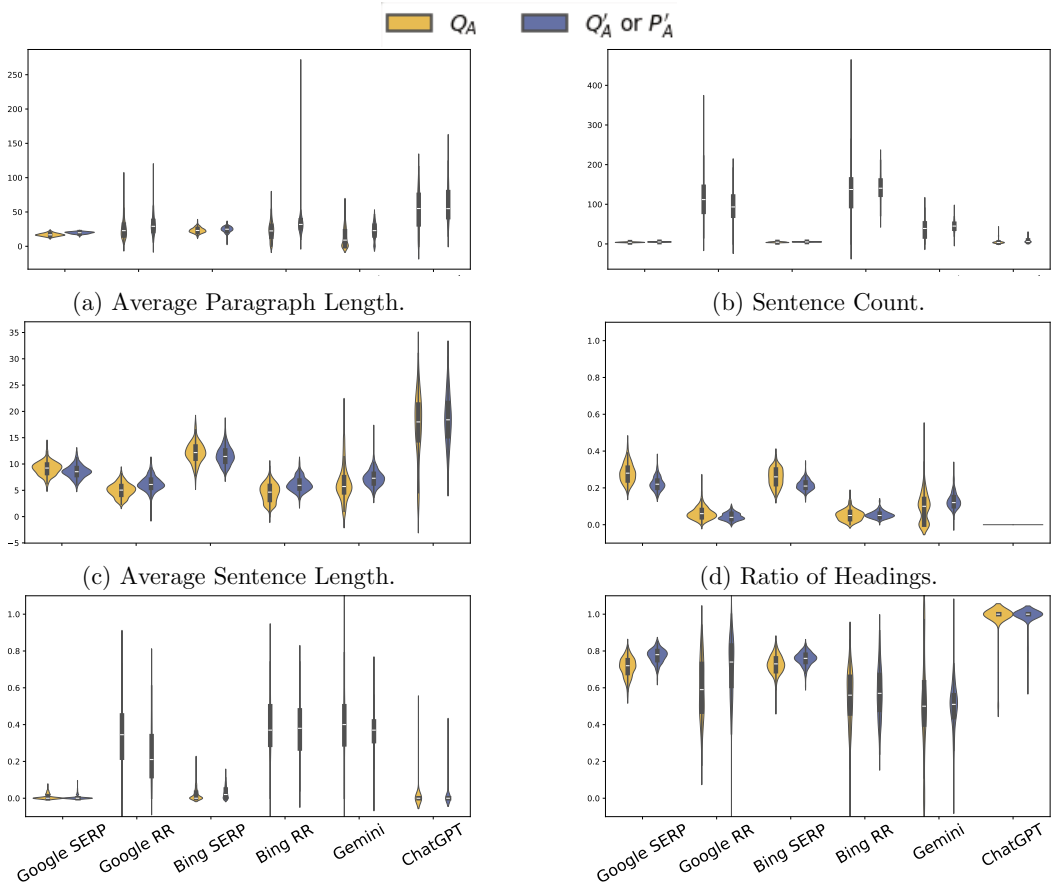
with AP(Q_A-G-RR), as reported in Table 7. Note that all reported comparisons are statistically significant unless stated otherwise.

From the Structure perspective, in terms of reading flow, AP(Q'_A-G-SERP) and AP(Q'_A-G-RR) have a higher ratio of paragraphs and a lower ratio of headings and list items than AP(Q_A-G-SERP) and AP(Q_A-G-RR), respectively. Based on indicators related to text succinctness, AP(Q'_A-G-SERP) and AP(Q'_A-G-RR) have higher average paragraph length than AP(Q_A-G-SERP) and AP(Q_A-G-RR) respectively. Yet we observe an inverse relation between the sentence count and average sentence length in the compared profiles, i.e., sentence count is lower but average sentence length is higher in AP(Q'_A-G-SERP) than in AP(Q_A-G-SERP); whereas in AP(Q'_A-G-RR) the sentence count is significantly higher and average sentence length significantly lower than in AP(Q_A-G-RR).

Based on Readability, AP(Q'_A-G-SERP) has a higher FRES score and lower CLI score than AP(Q_A-G-SERP). In fact, FRES and CLI scores of AP(Q'_A-G-SERP) meet the readability

criteria for autistic users, which implies that rephrasing the query resulted in Google SERP snippets that would actually be readable for autistic users. In the case of Google RR, AP($\mathtt{Q'_A}$-G-RR) also has a higher FRES score and lower CLI score than AP($\mathtt{Q_A}$-G-RR) but the higher scores don't meet the readability criteria.

In terms of Concreteness, the average concreteness in AP($\mathtt{Q'_A}$-G-SERP) is higher than in AP($\mathtt{Q_A}$-G-SERP) but the ratio of abstract words is also higher in AP($\mathtt{Q'_A}$-G-SERP) than in AP($\mathtt{Q_A}$-G-SERP). In case of Google RR on the other hand, AP($\mathtt{Q_A}$-G-RR) is more concrete on average with a lower ratio of abstract words than AP($\mathtt{Q'_A}$-G-RR); but the ratio of concrete words is higher in AP($\mathtt{Q'_A}$-G-RR) than in AP($\mathtt{Q_A}$-G-RR). From the perspective of Concreteness, it appears that the snippets in Google SERP are more concrete, on average, when the query specifies that the user is autistic, but the resources retrieved for such queries are less concrete on average compared to the resources retrieved when the queries do not state that the user is autistic.



(a) Average Paragraph Length.

(b) Sentence Count.

(c) Average Sentence Length.

(d) Ratio of Headings.

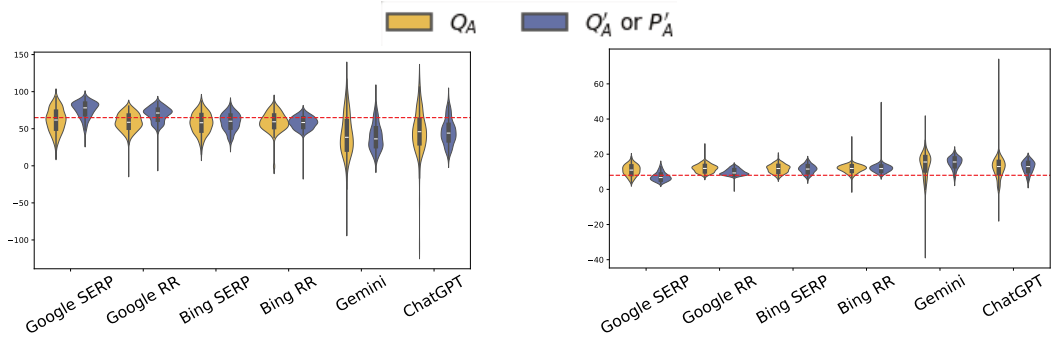
(e) Ratio of List Items.

(f) Ratio of Paragraphs.

Figure 17: Text structure scores for logs generated using $\mathtt{Q_A}$, $\mathtt{Q'_A}$, and $\mathtt{P'_A}$.

Bing. Comparing AP($\mathtt{Q_A\text{-}B\text{-}SERP}$) and AP($\mathtt{Q'_A\text{-}B\text{-}SERP}$) reported in Table 7, we identify several prominent differences between the profiles, and all reported comparisons are significant unless stated otherwise. Examining the text structure of the profiles in terms of text succinctness, AP($\mathtt{Q'_A\text{-}B\text{-}SERP}$) has a lower average paragraph length and a higher sentence count than AP($\mathtt{Q_A\text{-}B\text{-}SERP}$); but the average sentence length in AP($\mathtt{Q'_A\text{-}B\text{-}SERP}$) is lower than in AP($\mathtt{Q_A\text{-}B\text{-}SERP}$). In terms of reading flow, AP($\mathtt{Q'_A\text{-}B\text{-}SERP}$) has a lower ratio of headings and a higher ratio of paragraphs but a significantly higher ratio of list items than AP($\mathtt{Q_A\text{-}B\text{-}SERP}$). From the Readability perspective, AP($\mathtt{Q'_A\text{-}B\text{-}SERP}$) has a higher FRES score and lower CLI score than AP($\mathtt{Q_A\text{-}B\text{-}SERP}$). However, the difference in the readability indicator scores is not statistically significant. Based on the Concreteness perspective, AP($\mathtt{Q'_A\text{-}B\text{-}SERP}$) has a lower average concreteness and a higher ratio of abstract words, but a lower ratio of concrete words than AP($\mathtt{Q_A\text{-}B\text{-}SERP}$); however, only the difference in average concreteness holds true in practice. From these findings, we infer that rephrasing the query to state that the user is autistic impacts the structure and average concreteness of Bing SERP snippets but not the readability of the snippets.

In case of Bing RR responses, juxtaposing AP($\mathtt{Q_A\text{-}B\text{-}RR}$) with AP($\mathtt{Q'_A\text{-}B\text{-}RR}$) as reported in Table 7 reveals very few significant differences. Based on Structure, AP($\mathtt{Q_A\text{-}B\text{-}RR}$) has significantly lower average paragraph and sentence lengths than AP($\mathtt{Q'_A\text{-}B\text{-}RR}$). From Readability and Concreteness perspectives, AP($\mathtt{Q_A\text{-}B\text{-}RR}$) has higher FRES and lower CLI score, as well as higher average concreteness and lower ratio of abstract words than AP($\mathtt{Q'_A\text{-}B\text{-}RR}$), but only the differences in the ratio of abstract words holds true in practice. Based on these outcomes, we infer that resources retrieved by Bing for queries in $\mathtt{Q'_A}$ are less succinct and less concrete on average compared to the resources retrieved for the queries in $\mathtt{Q_A}$, but the readability of the resources retrieved for either query variation is similar on average.

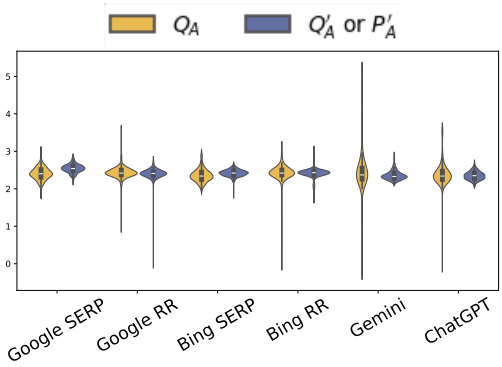


(a) FRES. (b) CLI.

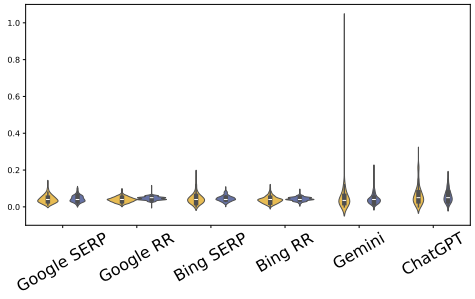
Figure 18: Text readability scores for logs generated using $\mathtt{Q_A}$, $\mathtt{Q'_A}$, and $\mathtt{P'_A}$. The dotted red line illustrates the minimum FRES score in (a) and maximum CLI score in (b) an IAS response must be considered readable for an autistic person.

Gemini. Turning to the accessibility profiles of LLM agents reported in Table 7, we first compare AP($\mathtt{Q_A\text{-}GEM\text{-}ANS}$) with AP($\mathtt{P'_A\text{-}GEM\text{-}ANS}$), and find that AP($\mathtt{Q_A\text{-}GEM\text{-}ANS}$) is notably more closely aligned to autistic users' accessibility needs than AP($\mathtt{P'_A\text{-}GEM\text{-}ANS}$) but not all differences hold true in practice. Based on Structure, AP($\mathtt{Q_A\text{-}GEM\text{-}ANS}$) has significantly

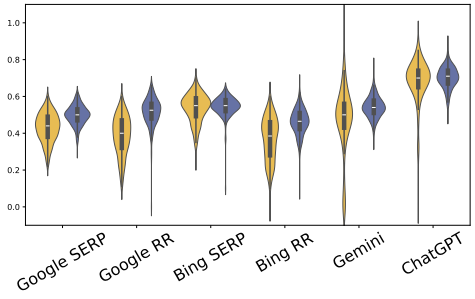
lower average paragraph and sentence lengths as well as significantly lower sentence count than AP($\texttt{P}'_\texttt{A}$-GEM-ANS). AP($\texttt{P}'_\texttt{A}$-GEM-ANS) has a higher ratio of headings and a lower ratio of paragraphs, but a lower ratio of list items than AP($\texttt{Q}_\texttt{A}$-GEM-ANS), although only the difference in the ratio of headings is statistically significant. In terms of Readability, AP($\texttt{Q}_\texttt{A}$-GEM-ANS) has a higher FRES score and significantly lower CLI score than AP($\texttt{P}'_\texttt{A}$-GEM-ANS) which implies that responses in $\texttt{Q}_\texttt{A}$-GEM-ANS are easier to read on average than the responses in $\texttt{P}'_\texttt{A}$-GEM-ANS. From the Concreteness perspective, AP($\texttt{P}'_\texttt{A}$-GEM-ANS) has a higher average concreteness than AP($\texttt{Q}_\texttt{A}$-GEM-ANS), but the difference is not significant. The ratio of abstract words, on the other hand, is significantly lower in AP($\texttt{Q}_\texttt{A}$-GEM-ANS) than in AP($\texttt{P}'_\texttt{A}$-GEM-ANS), from which we infer that rephrasing the query generally results in more abstract responses from Gemini.



(a) Average Concreteness.



(b) Ratio of Concrete Words.



(c) Ratio of Abstract Words.

Figure 19: Text concreteness scores for logs generated using $\texttt{Q}_\texttt{A}$, $\texttt{Q}'_\texttt{A}$ and $\texttt{P}'_\texttt{A}$.

ChatGPT. Examining the differences between AP($\texttt{Q}_\texttt{A}$-GPT-ANS) and AP($\texttt{P}'_\texttt{A}$-GPT-ANS) reported in Table 7 reveals that the scores pertaining to the reading flow of the responses remain unchanged. In terms of text succinctness, AP($\texttt{Q}_\texttt{A}$-GPT-ANS) has a significantly lower average paragraph and sentence lengths as well as significantly lower sentence count than AP($\texttt{P}'_\texttt{A}$-GPT-ANS). Even from the perspective of Readability, AP($\texttt{Q}_\texttt{A}$-GPT-ANS) has a higher FRES score and lower CLI score than AP($\texttt{P}'_\texttt{A}$-GPT-ANS), although the differences do not hold true in practice. In terms of Concreteness, AP($\texttt{Q}_\texttt{A}$-GPT-ANS) has a higher average concrete-

ness and a lower ratio of abstract words than AP($\mathtt{P'_A}$-GPT-ANS), but the difference in average concreteness is not statistically significant. Based on these findings, we infer that ChatGPT generates less succinct and less concrete answers on average for queries in $\mathtt{P'_A}$, compared to the answers it generates for queries in $\mathtt{Q_A}$.

5.3 Are the Responses of SE and LLM Agents More Accessible When Diagnosis Is Disclosed in a Query?

Based on our investigations juxtaposing responses produced by IASs to inquiries that do and do not explicitly state that the information needs are of an autistic user, we find that stating the user's condition in the inquiry does influence the type of resources retrieved by an SE as well as the answer generated by an LLM agent (§5.1). Comparing the accessibility profiles of each IAS generated based on responses for the two query variations (§5.2) reveals that from the perspective of text accessibility, SE responses are more prominently impacted compared to LLM agent responses. Particularly in the case of Google, the responses elicited using the rephrased queries are significantly easier to read with the SERP snippets even meeting the readability criteria for autistic users suggested in (Yaneva et al., 2015). However, both the snippets and the content in the retrieved resources are more verbose, and the structure of the responses, on average, is less suitable for an easy reading flow compared to the responses produced for the original queries. We observe a consistent trend across all the considered IASs that rephrasing the query to specify that the user is autistic results in responses that are less succinct than the responses produced for the original queries. The accessibility indicator scores captured in Figures 17, 18, and 19 reveal an interesting trend: the scores across all compared distributions are notably less varied for IAS responses to queries in $\mathtt{Q'_A}$ (or $\mathtt{P'_A}$) than the responses produced for queries in $\mathtt{Q_A}$. Overall, we conclude that although informing the IAS of the user's diagnosis does not improve the general accessibility of IAS responses for autistic individuals, the rephrasing leads to responses that are more consistent in linguistic style in terms of Structure, Readability, and Concreteness.

6 Discussion

The outcomes of our study reveal important implications about the accessibility of mainstream IASs in catering to autistic individuals' information needs, which we discuss in this section; we also outline the limitations of our study and suggest potential future research directions.

6.1 Text Accessibility of IAS Responses for Autistic Individuals

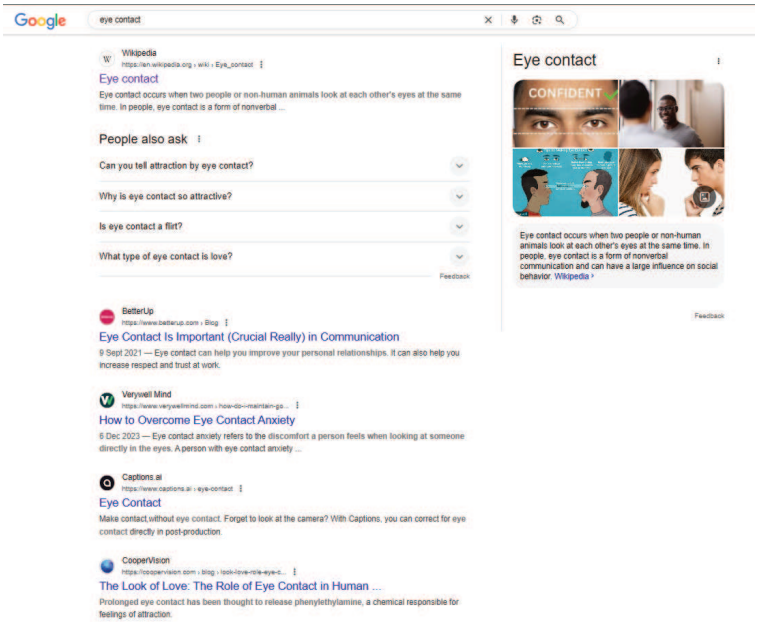
The results reported in Sections 4 and 5 highlight a potentially critical issue for autistic individuals who rely on popular IASs for information discovery. The accessibility profiles we generated of popular IASs to examine the text accessibility of IAS responses for autistic individuals' inquiries (RQ1) indicate that IAS responses to queries reflecting autistic individuals' common information needs are less accessible to the target audience than responses to inquiries from the general public. The issue is exacerbated when the autistic users' inquiry is related to their condition. Moreover, autistic individuals who are aware of their diagnosis may explicitly state their condition in their inquiry (Yechiam and Yom-Tov, 2021),

most likely with the expectation that the IAS—now aware of the target audience—would respond to the inquiry in a manner suitable for autistic individuals. However, based on the results of our explorations on whether the text accessibility of IAS responses improves when the inquiry is rephrased to state the user's diagnosis (RQ2), we find that only Google meets these expectations, but only from the readability perspective. In all other cases, explicitly stating that the user is autistic leads to more verbose responses in general that are not structured to facilitate an easy reading flow compared to the responses produced when the queries do not include the user's diagnosis. Overall, our findings suggest that, in terms of text accessibility, mainstream IASs react distinctively to autistic individuals' information needs. Moreover, IAS responses to autistic users' queries are less aligned to these users' expectations, compared to system responses to the general population's inquiries. This could hinder autistic user from *utilising* the IAS response to satisfy their information need, suggesting a concerning trend of mainstream commercial IASs potentially impeding autistic individuals' access to online information.

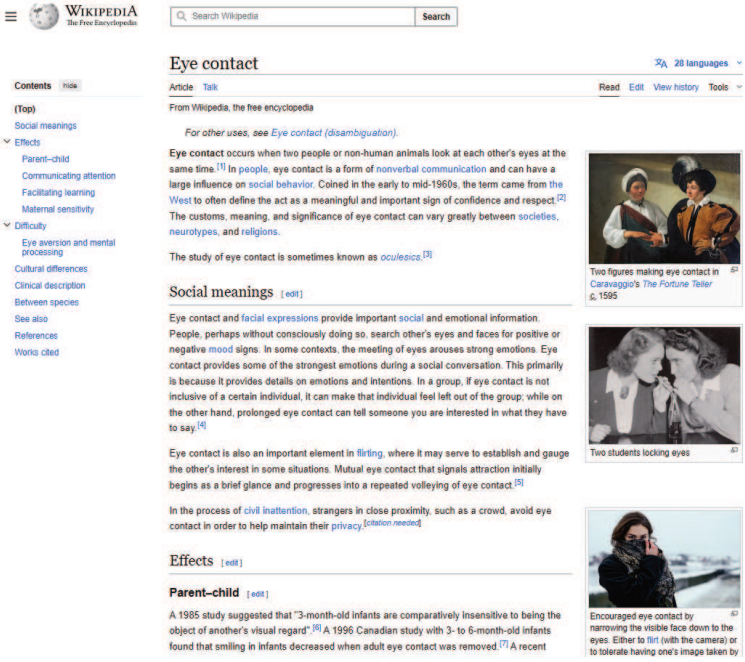
An unexpected outcome of our exploration was the impact of adding a diagnostic-disclosing phrase to the query on the linguistic *variability* of IAS responses. For an audience that struggles with adapting to unexpected changes as they prefer consistent and predictable interactions (WHO, 2023), the variation in the accessibility of IAS responses across query lengths and inquiry topics that we observed while examining the overall accessibility of IAS responses for autistic individuals (RQ1), can be anxiety-provoking. Specifying that the user is autistic in the inquiry to investigate its impact on the accessibility of IAS responses (RQ2) emerged as a solution to this issue. We found that the produced responses were notably more consistent in terms of accessibility across the rephrased inquiries compared to the accessibility of the responses for the original autistic individuals' inquiries.

6.2 Trade-Offs Across IAS

Our analysis highlights distinct text-based accessibility issues between SEs and LLM agents. LLM agents with their direct and succinct answers may appear to be an "autism-friendly" IAS, as they do not require the searcher to navigate through a list of resources, as is common in a traditional SE. Yet, from our analysis, we know that LLM agent responses, while succinct, are less concrete and harder to read than the web resources retrieved by an SE. This presents a trade-off, which we illustrate through an example. Consider an autistic user looking for information related to "eye contact". Entering this query in Google would yield a long list of snippets, as captured in Figure 20a. The user would then go through the list of short, easy-to-read snippets and choose a particular resource to explore further, like the Wikipedia article on eye contact illustrated in Figure 20b, for instance. The web resource would comprise a large amount of text compared to a snippet, but structured in a way that facilitates an easy reading flow. Moreover, based on the findings reported for RQ1, we could assume that the overall textual content would be much easier to read than the direct answer of an LLM agent. Using the same query on Gemini would yield a direct answer (as visualised in Figure 21). However, compared to the resources retrieved by Google, the answer would likely contain more complex sentences, so the user would have to spend extra effort to comprehend the text and extract the needed information. As mentioned previously, autistic individuals prefer consistency in their day-to-day interactions



(a) Top 5 ranked results in Google SERP.



(b) Excerpt from Wikipedia article on “eye contact”, the top-ranked result in Google SERP from Figure 20a.

Figure 20: Google response for an autistic user query: “eye contact”.

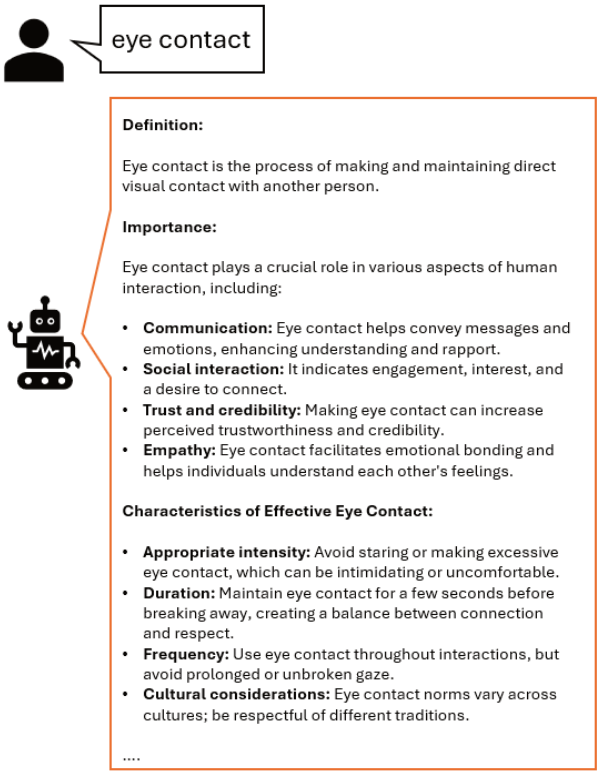


Figure 21: Excerpt from Gemini's generated response for an autistic user query: "eye contact".

(WHO, 2023). Therefore, the variation in the accessibility issues between SEs and LLM agents could hinder autistic users from using an IAS as they have grown accustomed to coping with the issues of the other system.

6.3 Designing for Accessibility

The individual profiles provide insights into the distinct strengths and limitations of each IAS in providing accessible responses to autistic individuals, which inspire strategies to enhance autistic users' overall online information-seeking process. Knowing that autistic individuals tend to scroll more on the SERP compared to non-autistic users when looking for information on an SE (Yechiam and Yom-Tov, 2021), it could be assumed that autistic users may not mind the extra effort of browsing through a SERP to find resources to answer their inquiry. Therefore, considering the easier readability and higher concreteness of SE responses, it may be useful to employ the better-structured response generation of LLM agents to further enhance the accessibility of SE responses for autistic users. For example, Gemini, in particular, generates responses structured to facilitate easier reading flow compared to the other IAS responses. Employing the LLM driving Gemini's response gen-

eration to restructure the verbose content in the resources retrieved by SE, as experimented in (Chhikara et al., 2024; Chang et al., 2023), could make it easier for autistic individuals to locate relevant components in an expansive web resource.

It is worth noting that SE responses, while easier to read than LLM responses, are yet to meet the readability criteria for autistic individuals. Stating the user is autistic in the inquiry leads to IAS responses that are even less aligned with autistic individuals' accessibility needs. Still, ChatGPT is capable of modifying its response to suit a specific need of a target audience, such as the reading level for a specified school grade (Murgia et al., 2023b). This could be utilised to simplify SE responses even for autistic individuals, by following the readability criteria suggested by Yaneva et al. (2015) and prompting ChatGPT to rephrase the SE response to suit the reading skill of an 8^{th} or 9^{th} grade student.

6.4 Limitations and Future Work

In the absence of publicly available data, we relied on synthetic queries to represent the information needs of autistic individuals. While our synthetic queries represent the plausible vocabulary and topics of interest common amongst autistic individuals, they may not represent other characteristics of how autistic users formulate their search queries, such as whether they formulate long or short queries, whether they formulate their queries using keywords only or opt for a more natural language approach, etc. Therefore, we advocate for the allocation of research efforts to collect representative query samples formulated by autistic individuals that can be used to examine whether their typical style of query formulation further influences the accessibility of IAS responses. Additionally, noting the difference in the readability of IAS responses for autistic users' inquiries directly related to ASD compared to more general inquiries (§4.3), future extensions of the study could also investigate the impact of autistic individuals' inquiry topic on the accessibility of the elicited IAS responses further based on more fine-grained categorisation of inquiry topics to reveal other trends. Regarding the experiments we conducted to answer RQ2, we explored the impact of including the diagnosis-reporting with a singular rephrasing strategy, but in the future, it may be beneficial to explore the potential of other known query rephrasing and prompt engineering strategies (e.g., Chen et al., 2021; Schmidt et al., 2023; Meskó, 2023; Xu et al., 2011) to yield IAS responses that are consistent in linguistic style but also accessible to autistic individuals.

Insights emerging from the accessibility profiles probed in our study showcase several potential accessibility limitations of popular IAS, even though we limited our analysis to the *textual* content of IAS responses to control the scope of our exploration. However, modern-day IAS are not limited to text modality. For instance, the online resources retrieved by an SE contain embedded videos, images, and sometimes even audio to keep the user engaged with the content. The latest versions of LLM agents are also capable of understanding and generating content with visual and audio elements (Islam and Moushi, 2024; Gemini Team Google et al., 2024). Multimodal content can be particularly beneficial for autistic individuals' access to online information—as long as each modality presents the same content—since it gives autistic individuals the freedom to interact with the same content in whichever modality they find the easiest to comprehend (WebAIM, 2025; Friedman and Bryen, 2007). Further, voice-controlled virtual assistants like Siri and Alexa, and video

platforms like YouTube, enable new search paradigms as interactions with such tools and platforms often lack a textual component. The diversification of online search paradigms and modalities raises other accessibility concerns for autistic individuals. For example, autistic individuals are very sensitive to loud noises (Williams et al., 2021; Landon et al., 2016), therefore sudden disturbing noises like fireworks and sirens in embedded videos can negatively impact the accessibility of the online content for autistic individuals (Britto and Pizzolato, 2016; Sitdhisanguan et al., 2012). Similarly, autistic individuals benefit greatly from visual stimuli during information search, but if the visual component is too abstract, then it could instead hinder a autistic person from understanding the content they are exposed to (Yaneva et al., 2015; Rezae et al., 2020). Considering autistic people's accessibility needs that go beyond reading a piece of text, it is important to examine the accessibility of popular IASs from a multimodal perspective. For this, COGA's proposed guidelines based on autistic individuals' challenges with comprehending content in non-textual modalities could be used in the future to extend the design of our experimental setup for generating accessibility profiles that capture the suitability of IAS responses for autistic individuals from a diverse perspective of web accessibility.

We examined the accessibility of the first response of different IASs to independent autistic individuals inquiries. As noted earlier, AI overviews were not consistently available at the time of the study and were therefore excluded from our analysis. However, as mainstream SEs increasingly integrate AI-generated summaries (or direct responses) at the top of SERPs Reid (2026, 2024), users are likely to encounter and engage with these summaries before looking at the list of retrieved results. With that in mind, it is important for future extensions of this work to also consider AI-generated overviews when assessing the accessibility of SE responses for autistic individuals. Furthermore, according to the information search process (**ISP**) model proposed by Kuhlthau (1991), a user goes through several stages when searching for information online. Starting from *initiation*, which is when the user realises they have a knowledge gap. This is followed by the *selection* and *exploration* stages, during which the user conducts a preliminary search to collect and explore resources that can help understand the general topic related to their knowledge gap. Our study captures the accessibility of mainstream IASs in supporting this preliminary search. But after the exploration stage, once the user has acquired an understanding of the general topic, they enter the *formulation* stage, wherein they may further refine their initial query to collect resources that are more aligned to their specific inquiry, followed by *collection* and *presentation* stages only after which a user's ISP is said to be complete. In the future, our study could be extended to probe user-IAS interactions at different stages of the user's ISP to get a more holistic overview of the accessibility of an IAS for autistic individuals.

The accessibility guidelines we use to inform the design of our experimental framework present a generalised overview of the challenges faced by autistic individuals, but one of the most characteristic traits of ASD is the variability in the abilities of autistic people (WHO, 2023). In the future, a complementary study could further validate the reliability of the accessibility profiles generated for the IAS by incorporating the perception of actual autistic users conducting self-directed online search tasks. Furthermore, considering the low diagnosis rate for ASD in several countries, especially amongst women (WHO, 2023), existing guidelines may not account for the distinct needs and expectations of these under-represented communities within a population that already is a minority in general. This

underscores the need for interdisciplinary research to broaden the understanding of ASD and its associated challenges to enable the creation of guidelines that represent the needs of *all* autistic individuals and not just an informed, stereotyped version of them.

7 Conclusions

In this work, we conducted an empirical exploration to gauge what kind of content autistic users are potentially exposed to when seeking information using mainstream IASs through the lens of Web Content Accessibility. In the absence of publicly available datasets, we constructed synthetic query collections using (i) key phrases from Reddit posts by self-reported autistic users—capturing topics, vocabulary, and phrasing common amongst autistic individuals—as proxy queries reflecting their information needs, along with (ii) a random sample from the Yahoo! Search Query Tiny dataset, to represent the information needs of the general population, respectively. We used these query collections to elicit responses from Google, Bing, ChatGPT, and Gemini. From the resulting IAS responses, we constructed an *accessibility profile* for each system. Rather than classifying a system as "good" or "bad" for autistic users, we use these profiles to probe IAS responses along three dimensions of text accessibility: structure, readability, and concreteness.

A nuanced exploration of the profiles (RQ1) reveals that IAS responses are less aligned to autistic individuals' accessibility requirements when catering to autistic users information needs, compared to when these systems respond to inquiries from the general population. The issue is exacerbated when an autistic user's inquiry is related to ASD. Rephrasing the autistic individuals' inquiries to explicitly include the user's condition in the inquiry to the IAS (RQ2)—an online search behaviour typical for diagnosed autistic individuals (Yechiam and Yom-Tov, 2021)—results in IAS responses that are even less accessible for this population. These findings suggest that autistic individuals relying on popular IASs for information discovery may struggle with finding the information they need to complete their information-seeking process. This highlights a critical limitation of popular IASs in catering to the information needs of autistic individuals and provides empirical evidence that could explain why autistic individuals so often feel frustrated while searching for information online (Gibson and Hanson-Baldauf, 2019).

Access to information is integral to a person's freedom of expression (UNESCO). Considering autistic individuals' tendency to give up on their search tasks when they are unable to find resources that fit their needs (Gibson and Hanson-Baldauf, 2019), we argue that the poor text accessibility of IAS responses could actively be *barring* autistic individuals from accessing crucial information online, which is a direct infringement of their fundamental rights as human beings. The European Accessibility Act also mandates all products and services—public or private—to comply with the accessibility requirements of people with disabilities (European Union, 2019). From both ethical and legal points of view, the outcomes of our study reveal a potentially concerning scenario of the accessibility of online information for autistic individuals, underscoring the need for researchers and developers alike to address the known issues as well as allocate efforts to recognise other limitations not only for autistic individuals but also other under-served communities, for which our experimental setup to generate accessibility profiles may serve as a useful reference.

Acknowledgments and Disclosure of Funding

The authors thank the reviewers and editor for their constructive feedback, which helped refine the final draft of the manuscript. This work was carried out without external funding or financial support from any funding agency, commercial entity, or other external organization.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Marwah Alaofi, Luke Gallagher, Dana Mckay, Lauren L. Saling, Mark Sanderson, Falk Scholer, Damiano Spina, and Ryen W. White. Where do queries come from? In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 2850–2862, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450387323. doi: 10.1145/3477495.3531711.

Garrett Allen, Ashlee Milton, Katherine Landau Wright, Jerry Alan Fails, Casey Kennington, and Maria Soledad Pera. Supercalifragilisticexpialidocious: Why using the "right" readability formula in children's web search matters. In *European Conference on Information Retrieval*, pages 3–18. Springer, 2022.

Mona Alzahrani, Alexandra L Uitdenbogerd, and Maria Spichkova. Impact of animated objects on autistic and non-autistic users. In *Proceedings of the 2022 ACM/IEEE 44th International Conference on Software Engineering: Software engineering in society*, pages 102–112, 2022.

American Psychiatric Association APA. *Diagnostic and statistical manual of mental disorders*. American Psychiatric Association Publishing, 2022.

Jesse G Bahrke. Mining for diagnostic criteria in the social behavior of people with asd: Using machine learning to explore aspects of reddit use. Master's thesis, Rosalind Franklin University of Medicine and Science, 2024.

Penny Benford and P Standen. The internet: a comfortable communication medium for people with asperger syndrome (as) and high functioning autism (hfa)? *Journal of Assistive Technologies*, 3(2):44–53, 2009.

Tiago Bianchi. Global search engine desktop market share 2024, May 2024. URL `https://www.statista.com/statistics/216573/worldwide-market-share-of-search-engines/`.

Roi Blanco, Harry Halpin, Daniel M Herzig, Peter Mika, Jeffrey Pound, Henry S Thompson, and T Tran Duc. Entity search evaluation over structured web data. In *Proceedings of the 1st international workshop on entity-oriented search workshop (SIGIR 2011), ACM, New York*, volume 14, pages 2181–2187, 2011.

Alexis Bosseler and Dominic W Massaro. Development and evaluation of a computer-animated tutor for vocabulary and language learning in children with autism. *Journal of autism and developmental disorders*, 33:653–672, 2003.

Kristen Bottema-Beutel, Steven K Kapp, Jessica Nina Lester, Noah J Sasson, and Brittany N Hand. Avoiding ableist language: Suggestions for autism researchers. *Autism in adulthood*, 2021.

Talita Britto and Ednaldo Pizzolato. Towards web accessibility guidelines of interaction and interface design for people with autism spectrum disorder. In *ACHI 2016: the ninth international conference on advances in computer-human interactions*, pages 1–7, 2016.

Marc Brysbaert, Amy Beth Warriner, and Victor Kuperman. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46:904–911, 2014.

Yancheng Cao, Yangyang He, Yonglin Chen, Menghan Chen, Shanhe You, Yulin Qiu, Min Liu, Chuan Luo, Chen Zheng, Xin Tong, et al. Designing llm-simulated immersive spaces to enhance autistic children's social affordances understanding in traffic settings. In *Proceedings of the 30th International Conference on Intelligent User Interfaces*, pages 519–537, 2025.

Emily Cary, Aparajita Rao, Erin Stephanie Misato Matsuba, and Natalie Russo. Barriers to an autistic identity: How rrbs may contribute to the underdiagnosis of females. *Research in Autism Spectrum Disorders*, 109:102275, 2023.

Centers for Disease Control and Prevention CDC. Signs and symptoms of autism spectrum disorder, 01 2024a. URL `https://www.cdc.gov/autism/signs-symptoms/index.html`.

Centers for Disease Control and Prevention CDC. About autism spectrum disorder, May 2024b. URL `https://www.cdc.gov/autism/about/index.html`.

Federica Cena, Noemi Mauro, Liliana Ardissono, Claudio Mattutino, Amon Rapp, Stefano Cocomazzi, Stefania Brighenti, and Roberto Keller. Personalized tourist guide for people with autism. In *Adjunct publication of the 28th ACM conference on user modeling, adaptation and personalization*, pages 347–351, 2020.

Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. Booookscore: A systematic exploration of book-length summarization in the era of llms. *arXiv preprint arXiv:2310.00785*, 2023.

Jia Chen, Jiaxin Mao, Yiqun Liu, Fan Zhang, Min Zhang, and Shaoping Ma. Towards a better understanding of query reformulation behavior in web search. In *Proceedings of the web conference 2021*, pages 743–755, 2021.

Garima Chhikara, Anurag Sharma, V Gurucharan, Kripabandhu Ghosh, and Abhijnan Chakraborty. Lamsum: A novel framework for extractive summarization of user generated content using llms. *arXiv preprint arXiv:2406.15809*, 2024.

Dasom Choi, Sunok Lee, Sung-In Kim, Kyungah Lee, Hee Jeong Yoo, Sangsu Lee, and Hwajung Hong. Unlock life with a chat(gpt): Integrating conversational ai with large

language models into everyday lives of autistic individuals. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400703300. doi: 10.1145/3613904.3641989. URL https://doi.org/10.1145/3613904.3641989.

Lei Chu, Hongyan Wu, and Yi Pan. Chatasd: A dialogue framework for llms enhanced by autism knowledge graph retrieval. In *Proceedings of the 15th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 1–8, 2024.

Meri Coleman and Ta Lin Liau. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283, 1975.

Cherie Courseault Trumbach and Dinah Payne. Identifying synonymous concepts in preparation for technology mining. *Journal of Information Science*, 33(6):660–677, 2007.

Antonina Dattolo, Flaminia L. Luccio, and Elisa Pirone. Webpage accessibility and usability for autistic users: a case study on a tourism website. In *International Conference on Advances in Computer-Human Interaction*, 2016a. URL https://api.semanticscholar.org/CorpusID:55048059.

Antonina Dattolo, Flaminia L Luccio, Elisa Pirone, et al. Web accessibility recommendations for the design of tourism websites for people with autism spectrum disorders. *International Journal on Advances in Life Sciences*, 8(3 & 4):297–308, 2016b.

Munmun De Choudhury, Scott Counts, Eric J Horvitz, and Aaron Hoff. Characterizing and predicting postpartum depression from shared facebook data. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 626–638, 2014.

Munmun De Choudhury, Emre Kiciman, Mark Dredze, Glen Coppersmith, and Mrinal Kumar. Discovering shifts to suicidal ideation from mental health content in social media. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, page 2098–2110, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450333627. doi: 10.1145/2858036.2858207.

Emani Dotch. Audiobuddy: An assistive technology for people with noise sensitivity and their care networks. In *Companion of the 2024 on ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '24, page 267–271, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400710582. doi: 10.1145/3675094.3678365.

Emani Dotch, Jazette Johnson, Rebecca W. Black, and Gillian R Hayes. Understanding noise sensitivity through interactions in two online autism forums. In *Proceedings of the 25th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '23, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400702204. doi: 10.1145/3597638.3608413.

B Driver and V Chester. The recognition and diagnosis of autism in women and girls: A literature review. *Advances in Autism*, 7(3):194–207, 2021.

Sukru Eraslan, Victoria Yaneva, Yeliz Yesilada, and Simon Harper. Do web users with autism experience barriers when searching for information within web pages? In *Proceedings of the 14th International Web for All Conference*, W4A '17, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450349000. doi: 10.1145/3058555.3058566.

Sukru Eraslan, Yeliz Yesilada, Victoria Yaneva, and Le An Ha. "keep it simple!": An eye-tracking study for exploring complexity and distinguishability of web pages for people with autism. *Universal Access in the Information Society*, 20(1):69–84, 2020. doi: 10.1007/s10209-020-00708-9.

Curts Eric. Chrome extensions for struggling students and special needs, Nov 2019. URL `https://www.controlaltachieve.com/2016/10/special-needs-extensions.html`.

European Union. Directive (eu) 2019/882 of the european parliament and of the council of 17 april 2019 on the accessibility requirements for products and services (text with eea relevance), 2019. URL `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32019L0882`.

Rudolf Flesch. How to write plain english, 07 2016. URL `https://web.archive.org/web/20160712094308/http://www.mang.canterbury.ac.nz/writing_guide/writing/flesch.shtml`.

Seraphina Fong, Alessandro Carollo, Giacomo Vivanti, Daniel S Messinger, Dagmara Dimitriou, and Gianluca Esposito. Autism spectrum disorders discourse on social media platforms: A topic modeling study of reddit posts. *Autism research*, 2025.

Mark G Friedman and Diane Nelson Bryen. Web accessibility design recommendations for people with cognitive disabilities. *Technology and disability*, 19(4):205–212, 2007.

Mark G. Friedman and Diane Nelson Bryen. Web accessibility design recommendations for people with cognitive disabilities. *Technology and Disability*, 19(4):205–212, 2008. doi: 10.3233/tad-2007-19406.

Uta Frith and Maggie Snowling. Reading for meaning and reading for sound in autistic and dyslexic children. *British journal of developmental psychology*, 1(4):329–342, 1983.

Gemini Team Google, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Gemini Team Google, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Amelia N Gibson and Dana Hanson-Baldauf. I want it the way i need it": Modality, readability, and format control for autistic information seekers online. *International Journal on Innovations in Online Education*, 2(4):1–12, 2019.

Kristen Gillespie-Lynch, Steven K Kapp, Christina Shane-Simpson, David Shane Smith, and Ted Hutman. Intersections between the autism spectrum and the internet: Perceived benefits and preferred functions of computer-mediated communication. *Intellectual and developmental Disabilities*, 52(6):456–469, 2014.

Maarten Grootendorst. Keybert: Minimal keyword extraction with bert., 2020. URL `https://doi.org/10.5281/zenodo.4461265`.

Francesca Happé and Uta Frith. The weak coherence account: detail-focused cognitive style in autism spectrum disorders. *Journal of autism and developmental disorders*, 36:5–25, 2006.

Francesca GE Happé. Central coherence and theory of mind in autism: Reading homographs in context. *British journal of developmental psychology*, 15(1):1–12, 1997.

Rukhshan Haroon and Fahad Dogar. Twips: A large language model powered texting application to simplify conversational nuances for autistic users. In *Proceedings of the 26th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–18, 2024.

Arno Hartholt, Sharon Mozgai, Ed Fast, Matt Liewer, Adam Reilly, Wendy Whitcup, and Albert" Skip" Rizzo. Virtual humans in augmented reality: A first step towards real-world embedded virtual roleplayers. In *Proceedings of the 7th international conference on human-agent interaction*, pages 205–207, 2019.

Amanda J Haskins, Thomas L Botch, Brenda D Garcia, Jennifer McLaren, and Caroline E Robertson. Gaze patterns modeled with a llm can be used to classify autistic vs. non-autistic viewers. *Journal of Vision*, 24(10):1190–1190, 2024.

Patricia Howlin and Philippa Moss. Adults with autism spectrum disorders. *The Canadian Journal of Psychiatry*, 57(5):275–283, 2012.

Chuanbo Hu, Wenqi Li, Mindi Ruan, Xiangxu Yu, Lynn K Paul, Shuo Wang, and Xin Li. Exploiting chatgpt for diagnosing autism-associated language disorders and identifying distinct features. *Research Square*, pages rs–3, 2024.

Raisa Islam and Owana Marzia Moushi. Gpt-4o: The cutting-edge advancement in multimodal llm. *Authorea Preprints*, 2024.

JiWoong Jang, Sanika Moharana, Patrick Carrington, and Andrew Begel. "it's the only thing i can trust": Envisioning large language model use by autistic workers for communication assistance. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2024.

Yi Jiang, Qingyang Shen, Shuzhong Lai, Shunyu Qi, Qian Zheng, Lin Yao, Yueming Wang, and Gang Pan. Copiloting diagnosis of autism in real clinical scenarios via llms. *arXiv preprint arXiv:2410.05684*, 2024.

MV Judy, U Krishnakumar, and AG Hari Narayanan. Constructing a personalized e-learning system for students with autism based on soft semantic web technologies. In *2012 IEEE International Conference on Technology Enhanced Education (ICTEE)*, pages 1–5. IEEE, 2012.

Tamar Kalandadze, Courtenay Norbury, Terje Nærland, and Kari-Anne B Næss. Figurative language comprehension in individuals with autism spectrum disorder: A meta-analytic review. *Autism*, 22(2):99–117, 2018.

Michelle R Kandalaft, Nyaz Didehbani, Daniel C Krawczyk, Tandra T Allen, and Sandra B Chapman. Virtual reality social cognition training for young adults with high-functioning autism. *Journal of autism and developmental disorders*, 43:34–44, 2013.

Gabriella Kazai, Paul Thomas, and Nick Craswell. The emotion profile of web search. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 1097–1100, 2019.

Lorcan Kenny, Caroline Hattersley, Bonnie Molins, Carole Buckley, Carol Povey, and Elizabeth Pellicano. Which terms should be used to describe autism? perspectives from the uk autism community. *Autism*, 20(4):442–462, 2016.

Jina Kim, Jieon Lee, Eunil Park, and Jinyoung Han. A deep learning model for detecting mental illness from user content on social media. *Scientific reports*, 10(1):11846, 2020.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. 1975.

N Koteyko. Understanding autistic adults' use of social media. *Proceedings of the ACM on Human-Computer Interaction*, 2023.

Carol Collier Kuhlthau. Inside the search process: Information seeking from the user's perspective. *J. Am. Soc. Inf. Sci.*, 42:361–371, 1991. URL `https://api.semanticscholar.org/CorpusID:14416802`.

Meng-Chuan Lai and Simon Baron-Cohen. Identifying the lost generation of adults with autism spectrum conditions. *The Lancet Psychiatry*, 2(11):1013–1027, 2015.

Jason Landon, Daniel Shepherd, and Veema Lodhia. A qualitative study of noise sensitivity in adults with autism spectrum disorder. *Research in Autism Spectrum Disorders*, 32: 43–52, 2016.

Monica Landoni, Maria Soledad Pera, Emiliana Murgia, and Theo Huibers. Inside out: Exploring the emotional side of search engines in the classroom. In *Proceedings of the 28th ACM conference on user modeling, adaptation and personalization*, pages 136–144, 2020.

Michael Liedtke and Matt O'Brien. Google launches gemini, upping the stakes in the global ai race, Dec 2023. URL `https://apnews.com/article/google-gemini-artificial-intelligence-launch-95d05d02051e75e20b574614ae720b8b`.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

Chu-Sui Lin, Shu-Hui Chang, Wen-Ying Liou, and Yu-Show Tsai. The development of a multimedia online language assessment tool for young children with autism. *Research in developmental disabilities*, 34(10):3553–3565, 2013.

Ling-Yi Lin and Pai-Chuan Huang. Quality of life and its related factors for adults with autism spectrum disorder. *Disability and rehabilitation*, 41(8):896–903, 2019.

Rachel Tsz-Wai Lo, Ben He, and Iadh Ounis. Automatically building a stopword list for an information retrieval system. In *Journal on Digital Information Management: Special Issue on the 5th Dutch-Belgian Information Retrieval Workshop (DIR)*, volume 5, pages 17–24, 2005.

Jill Locke, Eric H Ishijima, Connie Kasari, and Nancy London. Loneliness, friendship quality and the social networks of adolescents with high-functioning autism in an inclusive school setting. *Journal of research in special educational needs*, 10(2):74–81, 2010.

O Ivar Lovaas and Laura Schreibman. Stimulus overselectiv1ty of autistic children in a two stimulus situation. *Behaviour research and therapy*, 9(4):305–310, 1971.

David Mason, Helen McConachie, Deborah Garland, Alex Petrou, Jacqui Rodgers, and Jeremy R Parr. Predictors of quality of life for autistic adults. *Autism Research*, 11(8): 1138–1147, 2018.

Alexey N. Medvedev, Renaud Lambiotte, and Jean-Charles Delvenne. The anatomy of reddit: An overview of academic research. *Springer Proceedings in Complexity*, page 183–204, May 2019. doi: 10.1007/978-3-030-14683-2_9.

Bertalan Meskó. Prompt engineering as an important emerging skill for medical professionals: tutorial. *Journal of medical Internet research*, 25:e50638, 2023.

Dan Milmo. Google says new ai model gemini outperforms chatgpt in most tests, Dec 2023. URL `https://www.theguardian.com/technology/2023/dec/06/google-new-ai-model-gemini-bard-upgrade`.

Ashlee Milton and Maria Soledad Pera. Into the unknown: Exploration of search engines' responses to users with depression and anxiety. *ACM Trans. Web*, 17(4), 07 2023. ISSN 1559-1131. doi: 10.1145/3580283. URL `https://doi.org/10.1145/3580283`.

Emiliana Murgia, Zahra Abbasiantaeb, Mohammad Aliannejadi, Theo Huibers, Monica Landoni, and Maria Soledad Pera. Chatgpt in the classroom: A preliminary exploration on the feasibility of adapting chatgpt to support children's information discovery. In *Adjunct Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '23 Adjunct, page 22–27, New York, NY, USA, 2023a. Association for Computing Machinery. ISBN 9781450398916. doi: 10.1145/3563359.3597399. URL `https://doi.org/10.1145/3563359.3597399`.

Emiliana Murgia, Maria Soledad Pera, Monica Landoni, and Theo Huibers. Children on chatgpt readability in an educational context: Myth or opportunity? UMAP '23 Adjunct, page 311–316, New York, NY, USA, 2023b. Association for Computing Machinery. ISBN 9781450398916. doi: 10.1145/3563359.3596996. URL `https://doi.org/10.1145/3563359.3596996`.

Johan Nyrenius, Jonas Eberhard, Mohammad Ghaziuddin, Christopher Gillberg, and Eva Billstedt. The 'lost generation'in adult psychiatry: psychiatric, neurodevelopmental and sociodemographic characteristics of psychiatric patients with autism unrecognised in childhood. *BJPsych open*, 9(3):e89, 2023.

Irene M O'Connor and Perry D Klein. Exploration of strategies for facilitating the reading comprehension of high-functioning students with autism spectrum disorders. *Journal of autism and developmental disorders*, 34:115–127, 2004.

OpenAI. Introducing chatgpt, Nov 2022. URL `https://openai.com/index/chatgpt/`.

Elizabeth O'Nions, Irene Petersen, Joshua E.J. Buckman, Rebecca Charlton, Claudia Cooper, Anne Corbett, Francesca Happé, Jill Manthorpe, Marcus Richards, Rob Saunders, and et al. Autism in england: Assessing underdiagnosis in a population-based cohort study of prospectively collected primary care data. *The Lancet Regional Health - Europe*, 29:100626, 06 2023. doi: 10.1016/j.lanepe.2023.100626.

Allan Paivio. Dual coding theory, word abstractness, and emotion: a critical review of kousta et al.(2011). 2013.

Anna Y Park, Andy Jin, Jeremy Zhengqi Huang, Jesse Carr, and Dhruv Jain. Masksound: Exploring sound masking approaches to support people with autism in managing noise sensitivity. In *Proceedings of the 26th International ACM SIGACCESS Conference on Computers and Accessibility*, pages 1–12, 2024.

Michael Paul and Mark Dredze. You are what you tweet: Analyzing twitter for public health. In *Proceedings of the international AAAI conference on web and social media*, volume 5, pages 265–272, 2011.

Nikolay Pavlov. User interface for people with autism spectrum disorders. *Journal of Software Engineering and Applications*, 2014, 2014.

Maria Soledad Pera, Emiliana Murgia, Monica Landoni, Theo Huibers, and Mohammad Aliannejadi. Where a little change makes a big difference: a preliminary exploration of children's queries. In *European Conference on Information Retrieval*, pages 522–533. Springer, 2023.

Bertram O Ploog. Stimulus overselectivity four decades later: A review of the literature and its implications for current research in autism spectrum disorder. *Journal of autism and developmental disorders*, 40:1332–1349, 2010.

Cynthia Putnam and Lorna Chong. Software and technologies designed for people with autism: what do users want? In *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility*, Assets '08, page 3–10, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781595939760. doi: 10.1145/1414471.1414475. URL `https://doi.org/10.1145/1414471.1414475`.

Amon Rapp, Federica Cena, Guido Boella, Alessio Antonini, Alessia Calafiore, Stefania Buccoliero, Maurizio Tirassa, Roberto Keller, Romina Castaldo, and Stefania Brighenti. Interactive urban maps for people with autism spectrum disorder. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 1987–1992, 2017.

Vijayalaxmi N Rathod, RH Goudar, Anjanabhargavi Kulkarni, Dhananjaya GM, and Geetabai S Hukkeri. A survey on e-learning recommendation systems for autistic people. *IEEE Access*, 12:11723–11732, 2024.

Dora M Raymaker, Steven K Kapp, Katherine E McDonald, Michael Weiner, Elesia Ashkenazy, and Christina Nicolaidis. Development of the aaspire web accessibility guidelines for autistic web users. *Autism in Adulthood*, 1(2):146–157, 2019.

Elizabeth Reid. Generative ai in search: Let google do the searching for you, May 2024. URL `https://blog.google/products-and-platforms/products/search/generative-ai-google-search-may-2024/`.

Elizabeth Reid. A new era for ai search, May 2026. URL `https://blog.google/products-and-platforms/products/search/search-io-2026/#powerful-ai`.

Mortaza Rezae, Nigel Chen, David McMeekin, Tele Tan, Aneesh Krishna, and Hoe Lee. The evaluation of a mobile user interface for people on the autism spectrum: An eye movement study. *International Journal of Human-Computer Studies*, 142:102462, 2020. doi: 10.1016/j.ijhcs.2020.102462.

Esteban A Ríssola, Mohammad Aliannejadi, and Fabio Crestani. Mental disorders on online social media through the lens of language and behaviour: Analysis and visualisation. *Information processing & management*, 59(3):102890, 2022.

Sergio Rubio-Martín, María Teresa García-Ordás, Martín Bayón-Gutiérrez, Natalia Prieto-Fernández, and José Alberto Benítez-Andrades. Early detection of autism spectrum disorder through ai-powered analysis of social media texts. In *2023 IEEE 36th International symposium on computer-based medical systems (CBMS)*, pages 235–240. IEEE, 2023.

Ginny Russell, Sal Stapley, Tamsin Newlove-Delgado, Andrew Salmon, Rhianna White, Fiona Warren, Anita Pearson, and Tamsin Ford. Time trends in autism diagnosis over 20 years: a uk population-based cohort study. *Journal of Child Psychology and Psychiatry*, 63(6):674–682, 2022.

Siba Sankar Sahu, Debrup Dutta, Sukomal Pal, and Imran Rasheed. Effect of stopwords and stemming techniques in urdu ir. *SN Computer Science*, 4(5):547, 2023.

Alessia Sarica, Andrea Quattrone, and Aldo Quattrone. Introducing the rank-biased overlap as similarity measure for feature importance in explainable machine learning: a case study on parkinson's disease. In *International Conference on Brain Informatics*, pages 129–139. Springer, 2022.

Douglas C Schmidt, Jesse Spencer-Smith, Quchen Fu, and Jules White. Cataloging prompt patterns to enhance the discipline of prompt engineering. 2023. URL `https://www.dre.vanderbilt.edu/~schmidt/PDF/ADA_Europe_Position_Paper.pdf`.

Lisa Seeman and Michael Cooper. Cognitive accessibility user research, 12 2024. URL `https://www.w3.org/TR/coga-user-research/#autism`.

Dushyant Sharma, Rishabh Shukla, Anil Kumar Giri, and Sumit Kumar. A brief review on search engine optimization. *2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pages 687–692, 2019. URL `https://api.semanticscholar.org/CorpusID:199059492`.

Paul T Shattuck, Gael I Orsmond, Mary Wagner, and Benjamin P Cooper. Participation in social activities among adolescents with an autism spectrum disorder. *PloS one*, 6(11): e27176, 2011.

Safa Shubbar. *Advancing Autism Spectrum Disorder Diagnosis: A Phenotype-Genotype Co-Analysis and Retrieval-Augmented LLM Framework for Clinical Decision Support*. PhD thesis, Kent State University, 2025.

Karanya Sitdhisanguan, Nopporn Chotikakamthorn, Ajchara Dechaboon, and Patcharaporn Out. Using tangible user interfaces in computer-based training systems for low-functioning autistic children. *Personal and Ubiquitous Computing*, 16:143–155, 2012.

Amanda Hoover Samantha Spengler. For some autistic people, chatgpt is a lifeline, May 2023. URL `https://www.wired.com/story/for-some-autistic-people-chatgpt-is-a-lifeline/`.

Sandra Thom-Jones, Imogen Melgaard, and Chloe S Gordon. Autistic women's experience of motherhood: A qualitative analysis of reddit. *Journal of Autism and Developmental Disorders*, pages 1–11, 2024.

UNESCO. Right to information. URL `https://www.unesco.org/en/right-information`.

United Nations. Universal declaration of human rights, Dec 1948. URL `https://www.un.org/en/about-us/universal-declaration-of-human-rights`.

Cornelis J Van Rijsbergen. Information retrieval, 2nd edn. newton, ma, 1979.

Sharon Varghese, Emily Carlson, and Vinoo Alluri. Listening to the spectrum: Exploring music preferences and associations in autism spectrum disorder communities on reddit, 2024.

Mila Vulchanova and Valentin Vulchanov. Rethinking figurative language in autism: What evidence can we use for interventions? *Frontiers in Communication*, 7:910850, 2022.

World Wide Web Consortium W3C. Web accessibility initiative, 2019. URL `https://www.w3.org/WAI/`.

World Wide Web Consortium W3C. Cognitive and learning disabilities accessibility task force, 2021. URL `https://www.w3.org/WAI/about/groups/task-forces/coga/`.

Tao Wang, Monica Garfield, Pamela Wisniewski, and Xinru Page. Benefits and challenges for social media users on the autism spectrum. In *Companion Publication of the 2020 Conference on Computer Supported Cooperative Work and Social Computing*, pages 419–424, 2020.

WebAIM. Evaluating cognitive web accessibility with wave, 2025. URL `https://wave.webaim.org/cognitive`.

Kevin White and Shadi Abou-Zahra. Ian, data entry clerk with autism, 06 2024. URL `https://www.w3.org/WAI/people-use-web/user-stories/story-two/`.

Ryen W. White. Advancing the search frontier with ai agents. *Commun. ACM*, 67(9): 54–65, August 2024. ISSN 0001-0782. doi: 10.1145/3655615.

World Health Organization WHO. Autism, 03 2023. URL https://www.who.int/news-room/fact-sheets/detail/autism-spectrum-disorders.

Diane L Williams, Nancy J Minshew, and Gerald Goldstein. Further understanding of complex information processing in verbal adolescents and adults with autism spectrum disorders. *Autism*, 19(7):859–867, 2015.

Zachary J Williams, Jason L He, Carissa J Cascio, and Tiffany G Woynaroski. A review of decreased sound tolerance in autism: Definitions, phenomenology, and potential mechanisms. *Neuroscience & Biobehavioral Reviews*, 121:1–17, 2021.

Zheng Xu, Xiangfeng Luo, Jie Yu, and Weimin Xu. Mining web search engines for query suggestion. *Concurrency and Computation: Practice and Experience*, 23(10):1101–1113, 2011.

Yahoo! Search query tiny sample, 2009. URL https://web.archive.org/web/20250517143901/https://webscope.sandbox.yahoo.com/catalog.php?datatype=l.

Victoria Yaneva and Richard Evans. Six good predictors of autistic text comprehension. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 697–706, 2015.

Victoria Yaneva, Irina Temnikova, and Ruslan Mitkov. Accessible texts for autism: An eye-tracking study. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility*, ASSETS '15, page 49–57, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450334006. doi: 10.1145/2700648.2809852.

Eldad Yechiam and Elad Yom-Tov. Unique internet search strategies of individuals with self-stated autism: Quantitative analysis of search engine users' investigative behaviors. *Journal of Medical Internet Research*, 23(7):e23829, 2021.

Jinan Zeidan, Eric Fombonne, Julie Scorah, Alaa Ibrahim, Maureen S Durkin, Shekhar Saxena, Afiqah Yusuf, Andy Shih, and Mayada Elsabbagh. Global prevalence of autism: A systematic review update. *Autism research*, 15(5):778–790, 2022.

Huan Zhao, Zhaobo Zheng, Amy Swanson, Amy Weitlauf, Zachary Warren, and Nilanjan Sarkar. Design of a haptic-gripper virtual reality system (hg) for analyzing fine motor behaviors in children with autism. *ACM Transactions on Accessible Computing (TACCESS)*, 11(4):1–21, 2018.

Z. Kevin Zheng, Nandan Sarkar, Amy Swanson, Amy Weitlauf, Zachary Warren, and Nilanjan Sarkar. Cheerbrush: A novel interactive augmented reality coaching system for toothbrushing skills in children with autism spectrum disorder. *ACM Trans. Access. Comput.*, 14(4), October 2021. doi: 10.1145/3481642.

George Kingsley Zipf. *Human behavior and the principle of least effort: An introduction to human ecology*. Ravenio books, 2016.

Appendix A. Subreddits scraped for data collection

Name	Official Description	#Members
r/autism	Autism. Autism news, information and support. Please feel free to submit articles to enhance the knowledge, acceptance, understanding and research of Autism and ASD.	480K
r/aspergers	Aspergers. For those affected by Autism Spectrum Disorder, providing a space for support, discussion, and sharing experiences.	175K
r/AutismInWomen	AutismInWoman. A community for autists that are not cisgender males. We discuss the struggles, triumphs, and mundane life events that come with our autistic experience. We're LGBTQIA+ inclusive. TERFS not welcome. We are open to those who are self-suspecting autism, self-diagnosed as autistic, and formally diagnosed autistic (regardless of age), as we recognize the barriers around formal diagnosis and assessment. Please engage kindly, read our Rules and Wiki pages, and modmail if you have any questions ¡3	229K
r/AutismTranslated	AutismTranslated. If you think you might be autistic - or even if you're on The Quest, to figure out why life seems so much stranger and harder for you than it does for other people - then we made this space for you. It's one thing to read DSM diagnostic criteria or an Autism Parent's lamentations, and another to really hear us as we describe what it feels like to _be_ autistic. Welcome, and please feel free to ask questions! :)	64K
r/AutisticAdults	Adults on the Autistic Spectrum. For adults who are on the autistic spectrum, or think they might be on the autistic spectrum, regardless of diagnostic status. This is a relaxed, rules-light community, preferring discussion rather than memes and media. Non-autistic people are welcome if they are here for the benefit of autistic people (see Rule 1 before posting).	100K
r/AutisticPride	Autistic Pride. A bunch of proud autistic folks. No cure necessary—we're not diseased. We support societal reforms that accommodate both autistic and neurotypical communities equally.	56K

r/aspergirls	Life skills and healthy coping mechanisms for the ASD community. Aspergirls is a place to share advice and tips for topics related to autism and self-improvement. We help with INTERPERSONAL questions/struggles related to autism and life skills, personal growth, healthy coping mechanisms, etc. We are a support community for autists, please remain civil at all times when posting here. Thank you!	100K
r/AskAutism	Ask Autistics. A forum to hear the experiences of autistic individuals, and educate about neurodiversity, human rights concerns, and how to be a better human being both to autistic people and all citizens of the world.	10K
r/autism_controversial	Autism_controversial. This subreddit is for autistic people to talk about controversial topics revolving autism. People with close relations to autistic people are welcome to join the conversation, but please do not speak over actually autistic people.	585
r/AutismCertified	AutismCertified. A community for clinically-dxed autistics. Its purpose is to create a space where autistic representation is not diluted by those who are not officially confirmed to have ASD. — Disclaimer — - If you wish to speak on the autistic experience without having been professionally evaluated and found to have ASD, please go elsewhere.	3K
r/TrueEvilAutism	TrueEvilAutism. A venting and ranting subreddit intended for diagnosed autistics to express their experiences and frustrations	1K
r/AutisticPeeps	AutisticPeeps. This community gives a space for professionally diagnosed autistic people to thrive. Self-diagnosed people are explicitly not allowed to participate in this subreddit, but self-suspicion is fine. Self-suspicion is okay, but self-diagnosis is not.	7K
r/SpicyAutism	SpicyAutism. This is a subreddit for level 2/3/otherwise higher support needs autists, where we are the majority and feel understood and validated. This subreddit is a safe space for all autistic people, family members, doctors, teachers, etc., with the understanding that the priority is the comfort and inclusion of higher support needs autists and our experiences. Here you can ask questions, share experiences, talk about your interests, and more.	21K

r/austismlevel2and3	Autismlevel2and3. A place not overrun with level 1's telling us that Autism "isn't a disability"	1K
r/AutisticLiberation	AutisticLiberation. Welcome to our subreddit! We are a safe space for autistic leftists, regardless of background.	2K
r/Autism_Pride	Autism_Pride. A bunch of autistic leftists fighting against discrimination and celebrating our authentic selves.	4K

Table 8: List of subreddits and their official descriptions used for data collection.

Appendix B. Table of Abbreviations

Acronym	**Full Form**
ASD	Autism Spectrum Disorder
IAS	Information Access System
SE	Search Engine
LLM	Large Language Model
WAI	Web Accessibility Initiative
W3C	World Wide Web Consortium
WCAG	Web Content Accessibility Guidelines
COGA	Cognitive and Learning Disabilities Accessibility Task Force
SERP	Search Engine Result Page
RR	Retrieved Resources
ANS	Answer generated by an LLM in response to a user prompt

Table 9: List of abbreviations used in the manuscript.

Information Retrieval Research 2 (2026) 113-158 Submitted 12/25; Revised 06/26; Published 06/26

Beyond Hostility: Detecting Subtle and Overt Forms of Online Conflict with Multi-Objective Learning

Oliver Warke O.WARKE.1@RESEARCH.GLA.AC.UK
GAIR Lab – School of Computing Science, University of Glasgow
Glasgow, United Kingdom

Joemon M Jose JOEMON.JOSE@GLASGOW.AC.UK
GAIR Lab – School of Computing Science, University of Glasgow
Glasgow, United Kingdom

Jan Breitsohl JAN.BREITSOHL@GLASGOW.AC.UK
Adam Smith Business School, University of Glasgow
Glasgow, United Kingdom

Editor: Daniela Godoy

Abstract

Social networks have become a dominant and influential aspect of modern society, with numerous users engaging in observing, creating, and distributing content. The growth of content has led to user conflicts that include bullying, aggression, harassment, and threats. Consequently, recent research has been aimed at identifying and addressing these openly hostile forms of social conflict. However, in current studies, the less overtly hostile yet equally damaging types of conflict, including teasing, criticism, and sarcasm, have been overlooked. We utilise a previously introduced multi-class conflict dataset and extend its application through new modelling approaches for conflict classification, developing a robust multi-objective classification model to capture the full spectrum of conflict; from subtle tensions to open hostility, significantly advancing conflict detection capabilities. This innovative approach leverages class-based reward functions to improve model performance and is implemented by fine-tuning pre-trained transformer language models within a decision transformer framework. We also propose and evaluate multiple knowledge distillation strategies that compress large LLM teachers into efficient student models. These methods result in statistically significant increases in classification performances, leading to further evaluation of performance-cost trade-off. Our experiments on three datasets demonstrate superior recall, precision, F1-score, and accuracy compared to traditional state-of-the-art deep learning classifiers. Furthermore, we analyse class ambiguity and its impact on model performance as well as conducting thematic analysis on model misclassifications.

Keywords: Subtle and Overt Hostility, Online Conflict Detection, Social Media Behaviour Analysis, Knowledge Distillation, Transformer-Based NLP

1 Introduction

Social networks have become a primary avenue for interpersonal communication, with extensive research and evidence supporting its widespread use within society (Auxier and Anderson, 2021; Ortiz-Ospina and Roser, 2023; Bucher et al., 2018). Social media users communicate opinions, views, and experiences to a broad global audience. Consequently,

 WWW: https://irrj.org DOI: 10.54195/irrj.25366

increasing levels of negative behaviours and interactions have been observed on social media platforms (Akram and Kumar, 2017; Siddiqui et al., 2016; Berryman et al., 2018). Several factors contribute to the escalation of these behaviours, including anonymity, the absence of physical and social cues, and limited accountability (Gonzales, 2014). These detrimental conflict behaviours take many forms and are constantly evolving to present different threats. All of these factors lead to a toxic online environment, with severe consequences to the well-being of individuals and communities.

Existing research methodologies aim to detect overtly hostile and socially damaging behaviours, such as hate speech, aggression, and cyberbullying, primarily due to their evident nature (Fortuna and Nunes, 2018; Alkomah and Ma, 2022; Poletto et al., 2021). However, subtle negative behaviours, for example, sarcasm and teasing, have been shown to exert a significant detrimental influence on user well-being (Boroon et al., 2021; Kowalski, 2000; Wang et al., 2022; Ledley et al., 2006), since their perceived harm varies subjectively. We argue that lighter forms of conflict—such as criticism, sarcasm, and teasing—exist along a broader spectrum of negative behaviours. This spectrum also includes more severe actions like harassment and threats, which have received comparatively greater research attention. For example, all user behaviours depicted in Poletto et al. (2021), are located at the end of the spectrum. Nonetheless, little research focuses on detecting broader manifestations of conflicts, and hence it becomes imperative to investigate a diverse spectrum of adverse user interaction behaviours on social media, collectively referred to as "conflicts". We argue that lighter forms of conflict are not merely harmless interactions; they can independently exert significant negative effects on users and may also contribute to the escalation of exchanges into more severe and overtly hostile behaviours. Prior research has shown that negative interactions in online environments can intensify over time and develop into more hostile forms of communication (Cheng et al., 2015; Kowalski et al., 2014). This gap highlights the need for more nuanced modelling frameworks capable of capturing the full spectrum of conflict behaviours.

Detecting and addressing both the lighter and more extreme forms of negative behaviour along our conflict spectrum would allow for the behaviours to be addressed before escalating. Limiting detection to only the more overt forms of negative behaviour results in reactionary steps, rather than proactive prevention or monitoring. As discussed above, terminology within the domain tends to centre around more extreme negative behaviours. Bullying is one such commonly referenced term. However, whilst we do not diminish the impact of bullying, its core component of repetition is not necessary for harm to be inflicted on users. We argue that even a single instance of negative behaviour can significantly harm users, also supported by Wolak et al. (2007). These limitations highlight the need for nuanced conflict modelling frameworks and classifiers detecting a broader spectrum of individual conflict occurrences, which are scalable, efficient, and effective. This challenge is further compounded by recent shifts in platform-level moderation, where major social media sites are increasingly delegating content oversight to users. This raises urgent questions about how conflict detection can be scaled and automated.

It is also important to recognise how negative behaviours, negative user interactions, and negative content on social media are all becoming more commonplace and accepted. Whilst it is important to strike a balance between free speech and moderation of content, the acceptance that users will be subjected to large volumes of negative content is disturbing.

Popular sites such as Facebook and X (Twitter) are abandoning their more strict content policies in favour of more lax content controls. In January 2025, Mark Zuckerberg, the CEO of Facebook, announced that Facebook would be removing fact-checkers and moderators. Zuckerberg accepted that there would be more misinformation and hate on the platform but that it would be a worthwhile trade-off (Thompson and Conger, 2025; Duffy, 2025; Liv McMahon, 2025). In a video released on Meta, Zuckerberg stated that they were going to rely on users reporting content violations before they took action (Meta and Kaplan, Joel, 2025; Zuckerberg, 2025). This means that although many users can be exposed to harmful content on the platform, Meta will not take action unless users report it, despite having the ability to detect such content with their current system. X (Twitter) has also removed much of its content moderation team, which has also caused concern about the increase in harmful content on the platform (Alba and Wagner, 2023; Brewster, 2024; Sankaran, 2022). This in turn can only have damaging results for users on these platforms as they face increased levels of negativity, toxicity, and harmful content. Both Facebook and X have introduced "community notes" as a form of fact checking and content moderation, these notes place the burden on the users to moderate and fact-check the content on the platform, with little oversight from the platform and no training. This extra responsibility can only serve to expose users to even more negative content as they are forced to engage with the harmful content in an attempt to preserve a positive environment. This increasing relaxation of content policies not only reinforces the need to develop measures to detect negative user conflicts but also the need to raise awareness of the potential epidemic of negative social media spaces. Although it is always important to promote free speech and fight against censorship, it does not mean that users' mental and physical health should be ignored and moral principles abandoned.

With the increasing recognition of conflicts as a significant research area and a growing social issue (Fortuna and Nunes, 2018), there has been a rise in the availability of datasets and models focused on the phenomenon (Matamoros-Fernández and Farkas, 2021; Aboujaoude et al., 2015). Existing research has centred on binary hate datasets and published papers, with fewer multi-label and multi-class datasets available (Davidson et al., 2017; Elsafoury, 2020; Mollas et al., 2022). While these resources have contributed valuable insights, it is essential to note that current research datasets have limitations and do not encompass a wide range of social media conflicts. Hence, our work utilises a multi-class conflict dataset, developed using hate netnographies (Kozinets, 2002) and introduced in a previous work (Warke et al., 2024). This dataset serves as a benchmark for the evaluation of models specifically within a nuanced human behaviour task. When used in combination with other multi-class social media hate and negative behaviour datasets, it forms a thorough experimental setup for rigorously and systematically testing classification models across multiple domain adjacent tasks.

In multi-class classification, a substantial challenge arises due to the inherent interaction between distinct classes, leading to heightened complexities in distinguishing and isolating specific class instances. Particularly within multi-class scenarios, the occurrence of cross-talk between classes introduces a significant hurdle, consequently impacting the effectiveness of classification (Lango and Stefanowski, 2022). This challenge is worsened in social media scenarios due to the ambiguity, noisy and error-prone constitution of social media data (Bouazizi and Ohtsuki, 2019). In addition, the conventional strategy of

constructing test collections from social media, especially exploiting distant supervision, contributes to this situation's intricacies. In hate classification research, despite other researchers identifying the complexity and ambiguity of defining these behaviours, there has been no discussion on the profound impact of these issues on classification performance. To address these challenges, we evaluate three modeling strategies; Decision Transformer, hierarchical classification, and knowledge distillation. Each of these methods is designed to mitigate different aspects of multi-class ambiguity.

As a use case we investigate detecting conflicts effectively and experiment with social media data in prominent consumer brand communities. Brand communities are pages, groups, and timelines controlled by brands which provide the opportunity for consumers to interact with the brand and each other (Brogi, 2014). These communities embrace substantial global user cohorts, thereby rendering them of notable significance for consumer conflict research. Prior work within the marketing domain has underscored the challenges brands encounter in relation to conflicts. These challenges include but are not limited to; serious damage to brand reputation, harm to consumers well-being, withdrawal of users from the brand community, and consumer purchase intentions (Breitsohl et al., 2018; Wang et al., 2012).

Although the dataset is derived from brand community interactions, these environments represent an important and widely studied form of social media interaction. Brand communities host large and diverse groups of users who engage in discussions with both brands and other users, generating a wide range of communicative behaviours including disagreement, criticism, and conflict. As such, they provide rich and naturally occurring data for analysing online interactions. The behavioural categories captured in this dataset, such as teasing, criticism, sarcasm, trolling, harassment, and threats, are not unique to brand communities and have been documented across many social media platforms. Nevertheless, we acknowledge that the dataset originates from a specific online context. While the behaviours themselves are broadly observed in online discourse, further evaluation across additional platforms would strengthen the generalisability of the proposed modelling approaches.

As discussed, current approaches fail to capture the spectrum of conflicts, focusing research on methods that detect only extreme forms of hate. Therefore, in this work, we investigate three methods for increasing the baseline performance of a classification model: a Reward-based Model called ConflictDT, Hierarchical Classification Framework, and Knowledge Distillation. ConflictDT, draws inspiration from recent advances in sequential decision modeling, where class-level signals are used to guide model behaviour. It was first introduced in this prior work (Warke et al., 2024). The Hierarchical framework leverages relationship structure within conflict types, enabling coarse-to-fine classification across grouped behaviours. Finally, Knowledge distillation compresses large language models into efficient student models, allowing us to explore trade-offs between model size, supervision depth, and classification accuracy.

There are a few significant differences between ConflictDT and Knowledge Distillation approaches which makes for an interesting comparison; whilst both seek to improve the performance of a baseline model using additional signals, ConflictDT does so without the need for an expensive powerful LLM acting as a teacher. The evaluation of the two approaches centres on the relationship between performance increase and cost. Whilst other works do

touch upon this relationship comparing the baseline model with the enhanced model, few conduct a detailed performance-cost analysis of different enhanced model approaches.

We make the following contributions in this paper:

- We present a comparative evaluation with statistical analysis of three modeling strategies—ConflictDT, hierarchical classification, and knowledge distillation for improving baseline performance in nuanced conflict classification and other tasks.
- We adapt and evaluate a hierarchical classification model to exploit the relationship structure within conflict class groupings.
- We propose and compare three distinct knowledge distillation strategies for compressing LLMs into efficient student models; Soft Target Distillation, Feature Distillation using Probabilities, and Feature Distillation using Hidden State Embeddings.
- We conduct a performance-cost analysis of ConflictDT and KD approaches, quantifying trade-offs between accuracy and computational efficiency.
- We conduct a comprehensive empirical analysis of distillation hyperparameters across various benchmarks to evaluate performance, scalability, and efficiency.
- We analyse classifier performance beyond traditional metrics, conducting a popular social science technique, thematic analysis, to investigate misclassification trends.

Building on these contributions, we define the following research questions to structure our investigation:

- R.Q.1 How effective are the different classification models for improving multi-class classification? How do the different forms of knowledge distillation compare in performance?
- R.Q.2 Can the reward function within ConflictDT be used to guide model behaviour and improve classification performance?
- R.Q.3 How effective is the Hierarchical Model in capturing latent structure within conflict class groupings, and how does it compare to ConflictDT when distinguishing between lesser and more extreme conflict behaviours?
- R.Q.4 What patterns emerge in model misclassifications, and how do these align with social science theories of online communication?
- R.Q.5 How do distillation hyperparameters affect the performance, scalability, and efficiency of each knowledge distillation strategy?
- R.Q.6 How do model training costs compare with classification performance across baseline, ConflictDT, and KD-enhanced models?

2 Related Work

2.1 Hate and Conflict Classification

There are many works investigating hate detection that have developed various machine-learning models and explored different classification algorithms and feature representations. Advancements in deep learning have increased the effectiveness significantly (Minaee et al., 2021). Deep learning algorithms, including BERT (Devlin et al., 2019), LSTM (Hochreiter and Schmidhuber, 1997), and CNN (O'Shea and Nash, 2015), have demonstrated strong performance across different text classification tasks. Khan et al. (2020) proposed a CNN-based framework called HateClassify for labelling social media content, achieving competitive multiclass accuracy. Researchers have explored techniques to enhance the performance of these algorithms further (d'Sa et al., 2020; Naseem et al., 2019). Ali et al. (2023) developed a LSTM-GRU model, combining deep learning and graph analytics, which exhibits state-of-the-art performance over a six class hate speech dataset.

The Transformer architecture (Vaswani et al., 2017), commonly utilised in language modelling and recommendation tasks (Li et al., 2020; Sun et al., 2019a), has revolutionised text classification tasks. This is attributed to its capacity to capture relational information using self-attention mechanisms. Salminen et al. (2020) focused on detecting hate across multiple platforms and found models with BERT features superior. d'Sa et al. (2020) conducted hate speech classification using word embeddings and Deep Neural Networks (DNN), with the fine-tuning approach using BERT achieving significant improvements. Caselli et al. (2021) introduced HateBERT, a retrained BERT model that outperformed the base BERT model in detecting abusive language.

Another closely related line of research focuses on toxicity scoring frameworks used in large-scale content moderation systems. Tools such as Google's Perspective API estimate the likelihood that a comment will be perceived as toxic, abusive, or otherwise harmful, producing continuous toxicity scores that can be used to prioritise moderation decisions (Wulczyn et al., 2017b; Jigsaw). These systems are widely used in industry settings due to their scalability and simplicity, and toxicity scores are frequently used in research to analyse harmful language and evaluate moderation systems (Gehman et al., 2020; Vidgen et al., 2021b). However, toxicity scoring typically focuses on estimating overall harmfulness rather than distinguishing between specific behavioural categories. As a result, such approaches may struggle to capture the nuanced distinctions between different forms of conflict behaviours, such as teasing, sarcasm, criticism, or trolling. In contrast, this work models conflict as a structured multi-class classification problem, enabling finer-grained differentiation across a spectrum of antagonistic behaviours.

As evidenced by the focus of these influential works, existing detection methods tend to focus on overtly hostile behaviours, such as threats and harassment, while overlooking subtler forms like teasing, sarcasm, and criticism. This is primarily a function of the training collections used, which focus on high-severity language like hate speech and explicit abuse (Salminen et al., 2020; d'Sa et al., 2020; Caselli et al., 2021). This gap motivates the need for multi-class conflict classification frameworks that capture a broader behavioural spectrum.

2.2 Reward-Based Modelling in NLP

Recent work has explored the use of reward signals to guide model behaviour in NLP tasks. Decision Transformers (Chen et al., 2021) have shown promise in vision and language modelling domains (Wang et al., 2024, 2025; Lee et al., 2022; Janner et al., 2021), but their application to text classification remains under explored. Gontier et al. (2023) adapted Decision Transformers for interactive text environments, demonstrating that reward signals can be integrated with natural language inputs to influence model decisions. Our work builds on this foundation by applying class-based reward signals to guide multi-class conflict classification.

2.3 Hierarchical Classification Models

Hierarchical classification has been widely used in emotion and sentiment detection tasks. Hadjiharalambous et al. (2022) introduced a hierarchical approach within an emotion detection task seeks to first classify datapoints into higher level groups before classifying them into their final classes. The primary emotion task features emotions which fall into two higher level classes, positive and negative. The negative group features three emotions, whilst the positive only one. The paper also introduces secondary emotion detection tasks with multiple emotions in the positive group but does not achieve statistically significant results in these tasks. The novelty of applying the model within this task is therefore threefold. Firstly, the groupings within the conflict task, lesser and more extreme conflict behaviours, are less defined than positive and negative emotions. Therefore, it is interesting to see how the hierarchical model performs with more complex relationships in an adjacent domain. Secondly, although the hierarchical model performed better in the tasks involving multiple sub-groups within each higher level group, the results did not show statistically significant increases in performance. By using the hierarchical model within this paper, we expand upon and enhance the methodology, evaluation, and analysis in the previous hierarchical work.

2.4 Knowledge Distillation

Knowledge Distillation (KD), first formally introduced by Hinton et al. (2015), investigated transferring the "dark knowledge" from a large, high-performance "teacher" model to a smaller, more efficient "student" model. This significant research demonstrated that by training the student to mimic the softened output probabilities (often referred to as "soft targets") of the teacher, the student could achieve significantly improved performance compared to the baseline performance of the student model. This initial concept provided a framework for distilling complex decision boundaries, class relationships, and class characteristics that are not easily understood by the student model.

In its early stages, KD involved matching the classification outputs of the student and teacher models. FitNets (Romero et al., 2015) introduced the idea of using intermediate feature representations, termed "hints," to guide student models. This approach allowed the student to learn not just the teacher's final outputs, but also the internal feature representations of the teacher which contained rich detail. Patient Knowledge Distillation (Sun et al., 2019b) refined feature-based transfer by matching hidden representations across mul-

tiple layers of the teacher and student. This multi-layer alignment strategy often leads to improved convergence, more robust feature transfer, and better generalisation for the student model, as it allows the student to learn similar hierarchical representations to the teacher. Zagoruyko and Komodakis (2016) introduced attention mechanism distillation, where the student learns to mimic the teacher's attention patterns. These diverse approaches have expanded the KD domain and led to its application in many different fields.

The application of KD has been widely explored within transformer models. DistilBERT (Sanh et al., 2019) was a significant example of KD which drastically reduced the size of BERT while maintaining much of its performance. DistilBERT simplified the training pipeline to use output logits and intermediate embeddings with a specific loss function. TinyBERT (Jiao et al., 2020) introduced a multi-faceted approach for KD in transformers, utilising not only output logit distillation but also fine-grained transformer layer and hidden state distillation, demonstrating robust performance and the power of internal LLM representations. These works demonstrate that KD is a powerful and effective tool for improving the performance of smaller transformer models for deployment in real-world scenarios.

With the introduction of generative LLMs and their future iterations like GPT-3 (Brown et al., 2020), FLAN-T5 (Longpre et al., 2023), and LLaMA (Touvron et al., 2023), issues surrounding the scale and resource requirements of these models have become more prominent. Direct deployment is often too expensive or requires too much computational power, leading to many researchers exploring the different approaches to compressing LLMs via distillation and parameter-efficient tuning. Recent surveys, such as Yang et al. (2024), provide a taxonomy of LLM distillation approaches, categorizing them into output, feature, and hybrid-based methods. However, many of these models are evaluated on general tasks, with limited works examining nuanced tasks within specialised domains.

Feature-based knowledge distillation is common among models which are employed in semantically rich domains, where data consists of a multitude of complex factors such as nuance, context, and class relationships—e.g., conflict behaviour text classification. In the context of transformers, matching hidden state embeddings has been shown to be a strong KD approach (Wang et al., 2020b). More recently, representation-aware distillation has been applied to multitask and multilingual models (Turc et al., 2019).

Our work builds on these foundations by comparing multiple KD strategies in a multi-class conflict setting. Specifically, we evaluate soft loss distillation, feature distillation via probabilities, and feature distillation via hidden state embeddings. These methods are contrasted against ConflictDT and hierarchical classification to assess their effectiveness in capturing subtle behavioural distinctions. Despite recent progress, the best approach for distilling teacher model knowledge in nuanced classification remains an open question—one we address through systematic evaluation of loss signals and distillation weights.

3 Methodology

Problem Statement: We first define the problem as a text classification task with a dataset $D = (x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$ where each datapoint contains a text x and a corresponding true label y. The discrete action y taken by the model is the classification

of the data point into the set of classes. Each true label y belongs to a finite set of classes $C = 1, ..., N$ where N is the number of classes.[1]

3.1 ConflictDT classifier:

Our first approach, ConflictDT, reconceptualises multi-class conflict classification as a decision-transformer-based learning problem. Instead of treating classification as a static mapping between input and label, we model it as a sequence of decisions guided by reward signals. Building on the Decision Transformer framework (Chen et al., 2021), we design a distance-aware reward function that quantifies inter-class separation for each input text, thereby mitigating cross-talk between overlapping class boundaries and promoting more distinct and interpretable decision trajectories.

While the Decision Transformer has previously been applied in text-based reinforcement learning scenarios (Gontier et al., 2023), its use for supervised text classification remains largely unexplored. In this work, we extend the framework beyond traditional reinforcement learning by integrating class-based reward computation directly into the classification pipeline. This innovation allows us to exploit the DT's reward-conditioning capability without relying on an explicit RL environment, effectively transforming a supervised classification task into a structured decision-making problem. The resulting architecture combines raw textual features with class-distance rewards to achieve enhanced discrimination and reasoning coherence across classes.

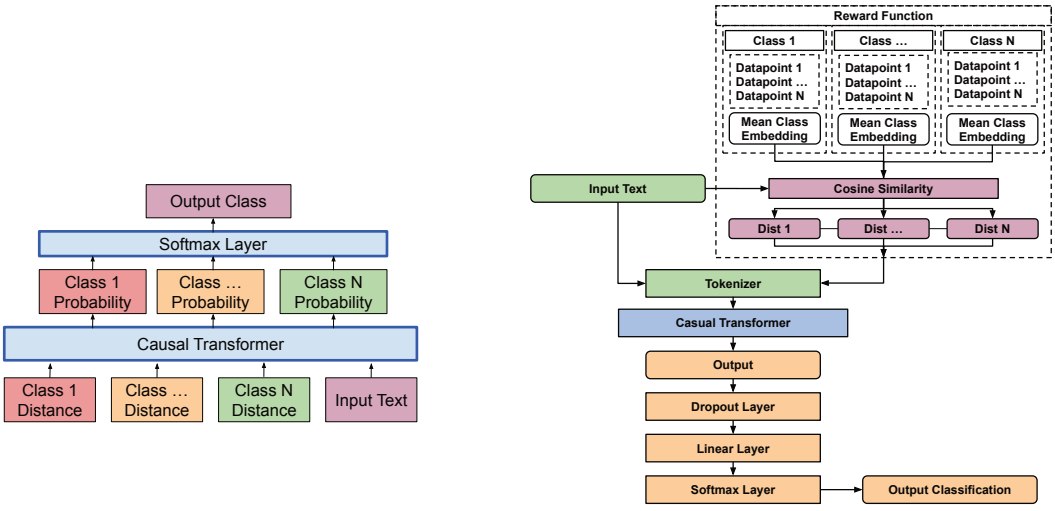


Figure 1: Model Diagrams of ConflictDT.

1. Code and dataset are available at https://github.com/OliverWarke/ConflictDT

3.2 Class Distance Reward Design

The focal point of this reward function is to optimise class distances with the overarching goal of enhancing classification performance. The core idea of our methodology is to improve upon classification performance when solely using raw datapoint text. We do this by providing the classifier with a quantitative measure of inter-class separation by creating a reward signal that quantifies the separation between the input text and each class. This aims to mitigate challenges due to inherent class ambiguities, inconsistency in short text lengths, and the contextual nature of nuanced human behavioural language, e.g. distinguishing between light teasing and more extreme trolling.

A visual representation of our proposed architectural framework is presented in Figure 3.1. Given an input text X, the input is first passed through a Sentence Transformer model, separate from the final classifier, to produce a dense text embedding e_x. This initial encoding is used exclusively to calculate the class distance, providing a robust, non-contextualised feature set.

As a basis for the reward function, we calculate the mean text embedding $\bar{y}_k$ for each class $k \in C$ from the training set, where C is the set of classes within the training set.

$$\bar{y}_k = \frac{1}{N_k} \sum_{i=1}^{N_k} y_k, i \tag{1}$$

The reward $r_{\bar{y}_k e_x}$ for each class $k \in C$ is computed using cosine similarity between the mean class embedding $\bar{y}_k$ and the input text embedding e_x:

$$r_{\bar{y}_k e_x} = CosSim(\bar{y}_k, e_x) \tag{2}$$

The resulting set of similarities $\{r_{\bar{y}_k} e_x\}_{k=1}^{N}$, i.e. class distances, forms the raw reward vector. We explored two distinct computational methods for employing the cosine similarity metric to quantify inter-class distances. The first method entailed calculating the mean vector for each of the class clusters (1) and gauging the cosine similarities between these mean vectors and text embedding. Conversely, the second method involved computing the cosine similarity between each possible pair of text embeddings of given text and elements of each class, then calculating an average cosine similarity. Upon evaluation, the first approach was deemed more efficient, thus becoming the chosen method implemented within our model.

The similarity/class distance values are then normalised and subsequently scaled to an integer range of 1 to 100. We adopted this scaling decision following research by Wallace et al. (2019), who investigate various NLP model's understanding of numbers. They found that transformer models do capture numerical features but some, notably BERT, struggle with decimal numbers. By scaling the distances we present them as integers making it more likely that BERT will understand them.

The scaled, normalised set of reward distances is converted into a string token, e.g. [Class Name: Scaled Reward Value,...] and prepended to the raw text input x to form the final input sequence S_{in}.

$$S_{in} = [RewardDistanceString|x] \tag{3}$$

Following these steps, the model follows the standard architecture of a deep learning transformer model, shown in Equation 4, with the exception of the input x which would normally be a textual embedding and is instead a combined text and reward feature embedding. The causal transformer structure of the DT model is simulated using BERT's self-attention mechanism to learn the relationship between the reward and the subsequent text.

$$z = \sum_{i=1}^{n} w_i * x_i + b, \quad a = \psi(z) \tag{4}$$

where w, weight, and b, bias, are trainable parameters, ψ is the Softmax activation function, action, a, is selecting one of the N dataset classes.

As depicted in Figure 1, the text and reward features are combined before going through the tokeniser and decision transformer layer, which further goes through a number of transformations. The output of the transformer model is first passed through a dropout layer to help prevent overfitting. The dropout layer randomly sets a number of the input tensor elements to zero depending on a set probability. Next, a linear layer, is used to produce the output logits, a vector of raw predictions for each class within the task.

Finally, a softmax layer is then used to produce the final probability distribution $\mathbf{a}$ over the $\boldsymbol{N}$ classes from the dataset using the final transformer output $\mathbf{z}$. This probability distribution is subsequently used to obtain the predicted class for the datapoint.

$$a = \psi(z) = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}} \tag{5}$$

Loss for the model is calculated using cross-entropy loss L_{CE} as follows:

$$L_{CE} = -\sum_{i=1}^{N} t_i \log(p_i) \tag{6}$$

for N classes, where t_i is the true label and p_i is the softmax probability for the ith class.

Although the Decision Transformer paper (Chen et al., 2021), which formed the foundation for this work, utilised the GPT-2 model (Radford et al., 2019), we ultimately decided to use the BERT model (Devlin et al., 2019) as the underlying transformer model within our research. This was due to BERT's superior performance in initial experiments, a decision reinforced by the findings of the experiments within this paper (see Tables 3, 4, 5, and 6).

In addition to this architectural change to the original work, we introduce a second alternative approach: the novel design of reward functions. Rewards can be modelled by various techniques tailored to suit distinct problem tasks, and we have experimented with different schemes, emphasizing the diversity of our approach. We explored numerous variations of reward functions based on distances during our experiments (section 4).

While the original DT paper (Chen et al., 2021) utilised transformer models within sequence modelling problems, our work presents a novel contribution by extending this framework to text classification. We further extended the framework to text classification by mapping states to sentences within each datapoint. By adapting the DT model to text classification, we enable the model to capture sequential and contextual information within the text datapoint. Subtle human conflict behaviours often depend on emphasis being placed on words or sentences occurring within a sequence. This adaptation and theoretical consideration represents a targeted extension of the original DT framework. To capture the information cues we explore two methods of dividing the text datapoints into sequences. Firstly, we divided the datapoints into sentences, with each state consisting of the previous states plus the next sentence. The first state would contain the first sentence, the second would contain the first two sentences; etc. The model would then provide a classification for each state, with the final state being the entire text datapoint. The second method was dividing the text into 3 sections with equal numbers of words. Then using the same method of combining the states at each timestep. The sequential classification models operate by inputting the current state's text and the rewards (if being used) to a BERT model. Then if timestep is >1. The CLS output of the model is combined with the CLS output at the previous timestep. This combined CLS embedding is passed through linear layers to produce a classification decision. This sequence modelling approach aimed to exploit the sequential nature of text, exposing the model to increased contextual information at each timestep, while incorporating the previous timesteps output layers.

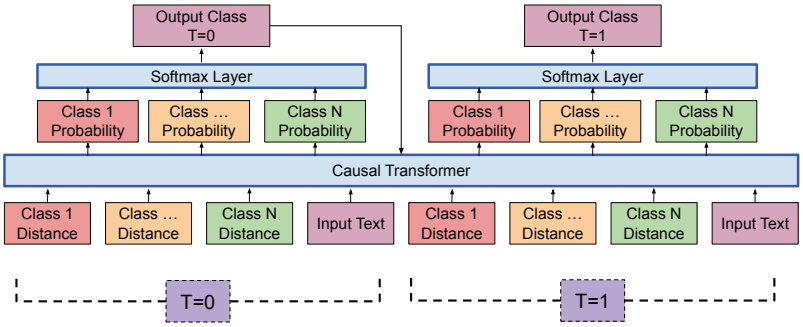


Figure 2: Architecture of Sequential ConflictDT.

3.3 Hierarchical Model

The conflict behaviours within our classification task naturally form a hierarchy. The behaviours exist within a range of light (teasing, criticism, sarcasm) to more extreme behaviours (trolling, harassment, threats). This division into parent groups and sub-classes suggests that a hierarchical classification strategy may be an effective means of exploiting these class relational groupings to improve classification performance.

We implement the Hierarchical model as described in Hadjiharalambous et al. (2022), with significant modifications to the methodology. We exploit the contrasting characteristics between the lesser and more extreme conflict behaviours. The Hierarchical version of

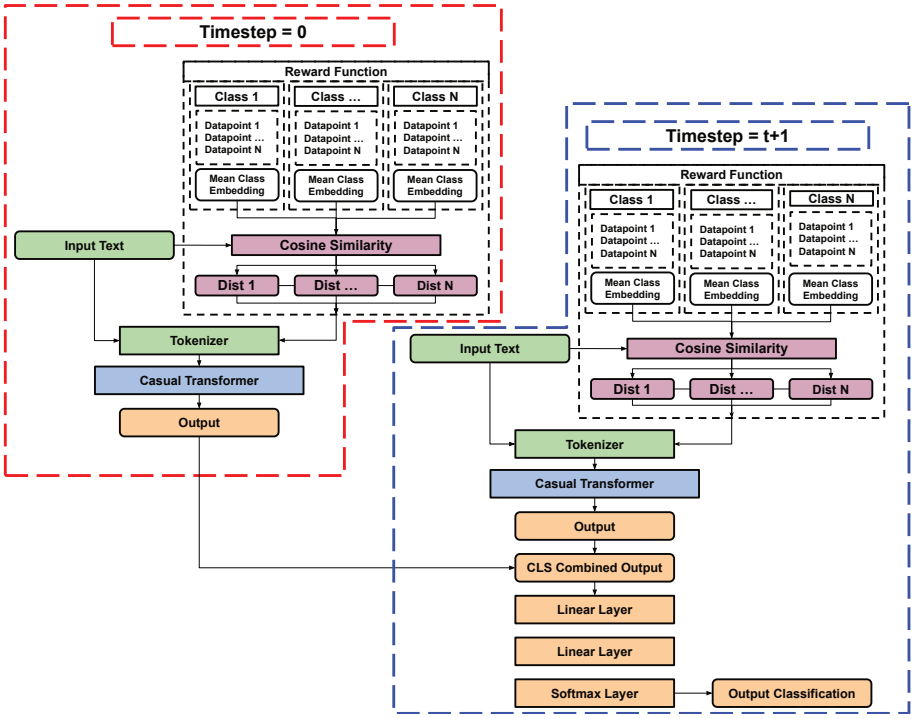


Figure 3: Sequential state construction for ConflictDT.

our classification task is displayed in Figure 4. In their work's main experiment Hadjiharalambous et al. use an emotion dataset featuring only one positive emotion, and therefore use only two separate loss functions; binary cross-entropy for higher level binary sentiment classification and negative log-likelihood for the multiclass negative emotion classification. This is then represented as:

$$L_T = L_1(Output_{Bi}) + L_2(Output_{Nmul}) \tag{7}$$

In our work we have multiple subclasses under each higher level group. We therefore use three separate loss functions; binary cross-entropy for higher level conflict severity classification, and two negative log-likelihood losses for the multiclass lesser and more extreme conflict subclass classification. These are then combined to produce the total model loss, consisting of the three sub-section losses.

$$L_T = L_1(Output_{Bi}) + L_2(Output_{Lesser}) + L_3(Output_{Extreme}) \tag{8}$$

In their methodology Hadjiharalambous et al. include a model architecture featuring a BiLSTM model, however, we opt to use a BERT model within the Hierarchical architecture for two reasons. Firstly, in their paper the BERT based Hierarchical model achieves the

best performance, and secondly we use a BERT model as a baseline model within the rest of the work so it allows for more robust performance comparisons.

This adaption from Hadjiharalambous et al. represents a more nuanced complex application of the hierarchical concept. Within emotions there are clearly defined positive and negative categories whilst conflict behaviours have more subtle differences. Additionally, we have modified the loss function to incorporate multiple sub-classes within the parent groups. Our hierarchical model therefore addresses a domain where the boundaries between parent groups are less defined and each sub-classification task is more complex. This tests the hierarchical model in a novel setting that has not previously been investigated.

Finally, we exploit the relationships between the two behaviour groups within the conflict task as a reward within the ConflictDT model, by utilising the performance of the hierarchical model we can compare and contrast two variations of classification model which seek to exploit the same structural relationships within the task in different ways.

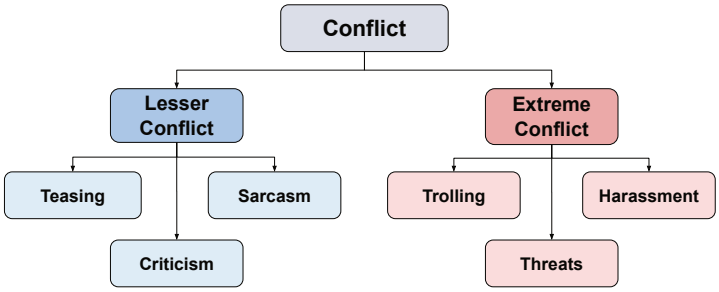


Figure 4: Hierarchy of the Conflict Classes

3.4 Distillation Methodologies

To explore the range of distillation signals available from teacher models, we evaluate three representative strategies that vary in granularity and depth of supervision:

- **Soft Target Distillation**: Leveraging softened output distributions from the teacher to guide the student. This is a widely used distillation strategy introduced by Hinton et al. Hinton et al. (2015).
- **Feature Distillation using Probabilities**: Introducing a regression head to align the student's probability vectors with those of the teacher. This has been shown in prior works to be competitive with soft target distillation Kim et al. (2021).
- **Feature Distillation using Embeddings**: Aligning the student's internal hidden states with those of the teacher modelRomero et al. (2015); Sun et al. (2019b).

Each training instance contributes two learning signals to the student model:

- **Hard label:** The ground truth class provided by human annotators.
- **Soft supervision:** Probability distributions or hidden-state representations extracted from the teacher model through prompted inference.

Using both signals follows standard knowledge distillation practice where the hard labels ensure alignment to the ground truth annotations whilst the soft supervision provides additional information from the teacher which is otherwise not available to the student.

The total loss for all distillation methods is computed as:

$$\mathcal{L}_{\text{total}} = (1 - \alpha) \cdot \mathcal{L}_{\text{CE}} + \alpha \cdot \mathcal{L}_{\text{Distill}} \tag{9}$$

where $\mathcal{L}_{\text{CE}}$ is the standard cross-entropy loss with hard labels, and α controls the influence of distillation-specific loss $\mathcal{L}_{\text{Distill}}$.

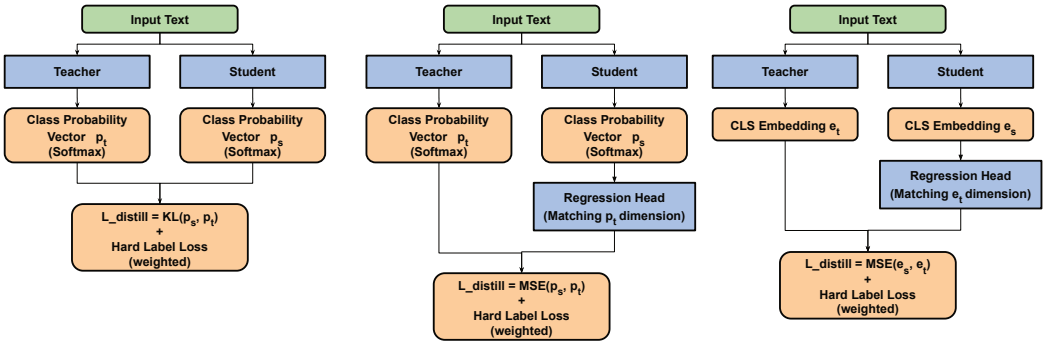


Figure 5: Figures showing Soft Target distillation, Feature Distillation, and Hidden State Embedding Distillation

Although these three distillation strategies are well-established in the knowledge distillation literature and are not novel in themselves, their comparative effectiveness has received limited evaluated in the context of fine-grained conflict classification. Our contribution lies in applying each method to this task, analysing how their differing supervision signals interact with the conflict taxonomy, and conducting an empirical study of their performance across a range of distillation weights. In our distillation experiments, the teacher model is **Meta-Llama-3-8B-Instruct**. Rather than fine-tuning the teacher as a task-specific classifier, we use it in prompted inference mode to generate supervision signals for the student model. All three distillation variants use the same teacher model to ensure comparability between supervision signals.

3.4.1 Soft Target Distillation

In this KD approach (Hinton et al., 2015), the teacher provides a softened probability distribution $\mathbf{p}_t$ over the class set. The student generates its own separate classification distribution $\mathbf{p}_s$.

$$\mathcal{L}_{\text{Distill}} = \text{KL}(\mathbf{p}_t \parallel \mathbf{p}_s) \tag{10}$$

This loss captures the fine grained similarities between classes, which are lost in one-hot ground truth labels. See Figure 3.4 for a schematic overview.

3.4.2 Feature Distillation (Probabilities)

Here, the student is tasked with directly regressing the teacher's class probability vector $\mathbf{p}_t$:

$$\mathcal{L}_{\text{Distill}} = \text{MSE}(\mathbf{p}_s, \mathbf{p}_t) \tag{11}$$

Unlike KL-divergence, MSE treats all errors equally, encouraging the student to match every probability with high fidelity (Kim et al., 2021). We implement this using a separate regression head parallel to the classification head. The architecture is shown in Figure 3.4.

For this conflict classification task, we chose MSE instead of KL-divergence as the conflict behaviours are semantically similar. KL-divergence emphasises relative differences between probabilities, which can amplify noise when teacher models are uncertain. MSE, however, encourages the student to match the teacher's output for all classes rather than just the relative ranking in KL-divergence. This is suitable to this task as the subtle distinctions between conflict behaviours mean that it is beneficial to the student model to be aware of all class probability not just the dominant class in the rankings.

3.4.3 Feature Distillation (Hidden State Raw Embeddings)

This approach transfers deeper knowledge by aligning internal representations. Specifically, we extract the final layer embedding vector $\mathbf{e}_t$ from the teacher and train the student to produce a matching embedding $\mathbf{e}_s$:

$$\mathcal{L}_{\text{Distill}} = \text{MSE}(\mathbf{e}_s, \mathbf{e}_t) \tag{12}$$

The student uses a regression head that maps its CLS token representation to the same dimensionality as the teacher's embedding.

3.5 Multi-class Conflict Dataset

Building on prior critiques of binary hate speech datasets (Davidson et al., 2017; Khan et al., 2020; Founta et al., 2018), we address the lack of coverage across the full spectrum of conflict behaviours by utilising a previously introduced multi-class conflict dataset derived from Facebook brand community interactions. While existing datasets predominantly focus on overtly hostile behaviours, this dataset captures a broader spectrum of conflict, including subtle forms such as teasing, sarcasm, and criticism.

The dataset originates from prior research investigating consumer conflicts within online brand communities (Breitsohl et al., 2018; Warke et al., 2024). It was constructed using a hate netnography methodology (Kozinets, 2015), which involves the systematic observation and analysis of naturally occurring online interactions. In the original study, social media interactions were collected across multiple brand communities over a sixteen-month period, enabling the identification of recurring patterns of consumer conflict behaviour. The resulting dataset was manually annotated by social science researchers specialising in online consumer interactions. The present work utilises this previously annotated dataset as a benchmark for evaluating machine learning approaches to conflict classification, while contributing new modelling techniques and experimental analysis.

In the original dataset creation process, the researchers adopted a dual-coding methodology to uphold the integrity of the annotation procedure, involving the active participation

of two social science researchers. The initial phase encompassed the first researcher's deductive identification of incidents involving consumer conflicts. An independent analysis of the data was then undertaken by a second researcher. Both researchers subsequently engaged in a comprehensive assessment of the reliability and applicability of their respective analyses. This deliberation was accompanied by extensive discussions to resolve any divergence in their interpretations. This rigorous and meticulous process culminated in identifying and classifying six distinct categories of conflicts, namely "Teasing", "Criticism", "Sarcasm", "Trolling", "Harassment", and "Threats". These categories span a spectrum of conflict severity and serve as the label space for all models evaluated in this study. The annotations used in this work therefore originate from the previously published dataset, and no additional manual annotation was conducted specifically for the present study. While the dataset itself was introduced in prior work, the contribution of the present study lies in the development and evaluation of novel modelling approaches for conflict classification. Specifically, this paper introduces the ConflictDT reward-based modelling framework, evaluates hierarchical classification strategies, and explores several knowledge distillation methods for improving multi-class conflict detection. The dataset therefore serves as a benchmark for systematically evaluating these modelling approaches rather than representing a newly introduced resource.

Stereotypical hate speech or general toxicity statements may appear across multiple conflict categories depending on their intent and severity. Mild or implicit stereotypes expressed through generalisations may align with the criticism class, whereas derogatory or demeaning stereotypes directed at individuals or groups would typically fall within harassment. In more extreme cases where stereotypes are used to justify intimidation or harm, they may overlap with the threats category. This flexibility reflects the broader aim of the conflict taxonomy to capture behaviours across a spectrum of antagonistic interaction rather than restricting classification to a single linguistic form.

As the six classes used within the task sit on a spectrum of conflict behaviours, theoretical considerations are required when dividing the classes into lesser and more extreme categories of conflict. Teasing, Criticism, and Sarcasm were all selected as less hostile forms of conflict whilst trolling, harassment, and threats were denoted as the more hostile forms of conflict. Sarcasm and teasing can both be considered lighter forms of conflict as there is a significant presence of humour in both, although they still have the potential to cause harm (Wang et al., 2022). Research such as Keltner et al. (2001) supports the similarity of sarcasm and teasing, but reasserts that they are two separate entities. Criticism borders the line between light and severe conflict; although it can be delivered in a constructive manner, it can also be received as a direct insult. Carlson (2016) and Xia (2013) highlight the impact of criticism in different communities within social media. Ultimately, criticism was seen as a socially acceptable form of interaction which, although not always the priority of the perpetrator, can result in harm to the recipient. Alternatively harm is almost exclusively the motive of those conducting trolling, harassment, and threats. These three classes have a clear motivation to cause severe distress to the recipient and therefore are definitively placed within the extreme forms of conflict.

While the netnographic process yielded rich annotations, certain classes—particularly Sarcasm, Threats, and Trolling—were under-represented. To ensure balanced evaluation, we supplemented these categories using external datasets with matching behavioural charac-

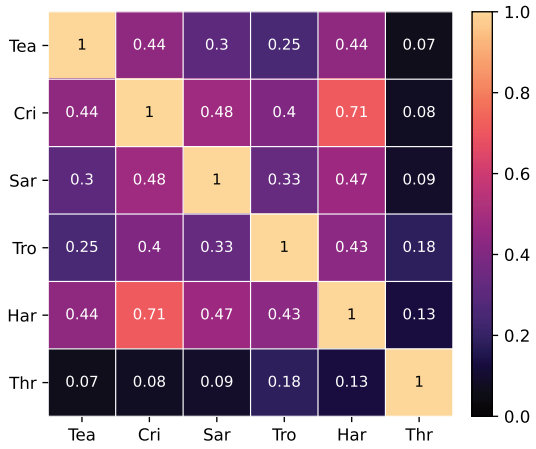


Figure 6: Heatmap showing class cosine similarity.

teristics. The open source datasets chosen were Khodak et al. (2018) for Sarcasm, Wulczyn et al. (2017a) for Threats, and Aggarwal et al. (2020) for Trolling. These external datasets were chosen as the characteristics of the required behaviours within them matched those within this paper. Other researchers (Fortuna et al., 2018; Salminen et al., 2020) investigate this process and find that the practice retains dataset robustness and in some cases can increase task performance. This hybrid construction approach ensures both conceptual integrity and sufficient class representation for training and evaluation. Final conflict dataset statistics are shown in Table 2.

3.6 Dataset Characteristics

To examine the robustness of the multi-class conflict dataset utilised in this paper, we conduct an evaluation of its structure, class balance, and semantic distinctiveness. We examine class size distributions, lexical and semantic features, and inter-class similarity using cosine distance metrics, as outlined in the Methodology. The goal is to assess the dataset's structural integrity and its suitability for training nuanced classifiers. This analytical review helps identify trends and features that could inform model design and evaluation. We also highlight its strengths and limitations to support future research and replication.

A key strength of the conflict dataset is that it encompasses a range of negative behaviours spanning subtle and overt forms of online conflict. As highlighted by Bianchi et al. (2022), there is a scarcity of research on nuanced hate and conflict; this paper addresses that gap by applying a multi-class conflict dataset grounded in established marketing theory and annotated by social science researchers with expertise in online conflict.

Due to the task's complex and real-life nature, we investigated semantic similarity between the six classes using a heatmap (Figure 6), and observed an average inter-class similarity of 0.32. Whilst similarities between some conflicts are low—e.g., "Threats" and the other

Class	Chars	Words	% Stop Words	Number Sentences
Teasing	78.5	14.6	0.31	1.6
Criticism	232.9	42.7	0.41	2.9
Sarcasm	58.9	10.7	0.32	1.1
Trolling	105.4	18.6	0.32	1.6
Harassment	130.2	23.9	0.35	2.2
Threats	275.7	50.6	0.31	5.5

Table 1: Table showing mean class characteristics within the conflict dataset

classes—there is less distinction between others, such as "Criticism" and "Harassment", which had a high similarity score of 0.71. This overlap poses a challenge for classification, particularly given the dataset's modest size. Higher similarity between classes results in greater difficulty for classifiers, as the decision boundaries become more ambiguous.

Text brevity is another key factor influencing classification performance. We conducted an analysis of textual features (Table 1), finding that the average character length and word count across all classes were just 147 characters and 26.85 words, respectively. Although prior research has established a correlation between shorter texts and reduced classification performance (Song et al., 2014), the conflict dataset's brevity accurately reflects the domain's characteristics and the nature of real-world social media interactions. Nonetheless, developing robust methods to handle short texts and semantic overlap remains essential.

By generating a class similarity matrix, we delve into the dataset's composition beyond class size, gaining insights that inform both evaluation and the design of our custom reward functions. For instance, we observe that milder conflicts tend to exhibit greater semantic similarity than severe ones. A particularly high level of similarity was found between the "Criticism" and "Harassment" classes. This may reflect behavioural overlap, as previous works (Kim et al., 2022; Jhaver et al., 2018) have investigated how criticism can be perceived as harassment, and how one may evolve into the other depending on context and actor intent. These findings directly inform the reward function design for ConflictDT involving class-level and group-level separation strategies.

Despite its merits, the dataset has limitations. While its size is comparable to other works (Sanh et al., 2019), it remains relatively small compared to large-scale hate speech datasets (Davidson et al., 2017; Vidgen et al., 2021b). However, state-of-the-art deep learning models have demonstrated strong performance on small datasets (Sanh et al., 2019; Caselli et al., 2021), and our experiments confirm that nuanced classification is still feasible. Another drawback is class imbalance: due to the nature of social media discourse, not all conflict types are equally represented. This imbalance must be accounted for in model training and evaluation.

Lesser Conflicts		More Extreme Conflicts	
Class	Datapoints	Class	Datapoints
Teasing	208	Trolling	1089
Criticism	698	Harassment	1098
Sarcasm	577	Threats	482

Table 2: Table showing conflict dataset class sizes

4 Implementation and Baselines

4.1 Implementation

All models were trained for four epochs using a learning rate of 2e-5 and the same Cross Entropy Loss Function. For a classification task, the number of output features is equal to the number of classes. We optimised models using AdamW (Loshchilov and Hutter, 2017). For the baseline models, all hyperparameters were used from the published papers, trained using the same setup and fine-tuned on the same training, validation, and test splits from each dataset. The train, validation, and test splits for all datasets were 80%, 10%, and 10% respectively. For the sentence transformer, we used the all-mpnet-base-v2 (Reimers and Gurevych, 2021)

4.2 Metrics and Statistical Tests

To evaluate classifier performance, we employed four key metrics: Accuracy, Precision, Recall, and F1-Score. Given the imbalanced nature of the conflict dataset, both Accuracy and F1-Score were particularly important for assessing model robustness. To further validate the significance of the observed performance differences, we conducted statistical testing using a paired two-tailed t-test. To obtain the performance metrics for each model, we performed 5-fold cross-validation and report the average macro results. When comparing model performance for each dataset we calculated the t-value between the enhanced performance models and the base BERT model. The t-value was then used to calculate a p-value which was evaluated against alphas of $\alpha = 0.05$ and $\alpha = 0.10$ to account for borderline effects in small sample comparisons.

4.3 Baselines

The following baselines were used:

- BERT: due to proven performance in classification tasks, especially within social media text problems (Wang et al., 2020a; Gupta et al., 2020). For the BERT model, we used the BERT-Base uncased pre-trained model with 12 layers, 12 heads, 768 hidden size, and 110M parameters.

- Flan-T5: A generative language model which can be fine tuned for text classification. Edwards and Camacho-Collados (2024) evaluate Flan-T5 for a multitude of text classification problems, showing strong performance metrics. We use the Flan-T5 base

model with 248M parameters uploaded to the Hugging Face repository by the Google team (Chung et al., 2024; et al., 2023).

- GPT-2 : demonstrated effectiveness in text classification tasks and its use in Decision Transformer work by Minaee et al. (2021); Balkus and Yan (2024); Chen et al. (2021). For the GPT-2 model, we used the default GPT-2 model with 12 layers, 12 heads, 768 hidden size, and 117M parameters.

- HateBERT: (Caselli et al., 2021), focusing on abusive language detection, specifically offensive, abusive and hateful language. HateBERT features intensive pre-training on social media comments before being deployed for fine-tuning domain-specific tasks. For HateBERT we used the default model provided by the authors with 12 layers, 12 heads, 768 hidden size, and 110M parameters.

- DistilBERT: a lightweight variation of the base BERT model, has been proven to be an excellent competitor to the traditional BERT model, with Sanh et al. (2019) stating: "it is possible to reduce the size of a BERT model by 40%, while retaining 97% of its language understanding capabilities and being 60% faster". We elected to benchmark DistilBERT for its successful use by Mutanga et al. (2020) within a hate classification task. For the DistilBERT model we used the default DistilBERT model with 6 layers, 768 hidden size, 12 heads, and 66M parameters.

Generative models, Flan-T5 and GPT-2, were also included to assess whether sequence-to-sequence architectures offer advantages in nuanced classification tasks. These baselines provide a diverse set of architectures commonly used in text classification research, including encoder-based models (BERT, DistilBERT, HateBERT), generative transformer models (GPT-2), and instruction-tuned sequence-to-sequence models (Flan-T5). This allows comparison of the proposed approaches across different transformer architectures while maintaining models that are widely used and reproducible in the research community.

The focus of this work is on evaluating modelling strategies for conflict classification rather than improvements driven purely by increasing model size. For this reason, we prioritise widely adopted transformer baselines used in antagonistic and negative behaviour research that allow consistent fine-tuning and comparison across experiments. Larger instruction-tuned LLMs such as LLaMA or GPT-3.5 have recently demonstrated strong performance on a variety of tasks and would therefore represent a natural direction for future evaluation of the proposed methodology.

4.4 Experimental Models

- ConflictDT: A variation of a BERT model with a class-based reward function based on the Decision Transformer Framework. Variations of this model are tested such as various reward functions and a sequential model.

- Hierarchical BERT: A BERT-based hierarchical model which exploits the relationship between lesser and more extreme conflict behaviours. Features a novel loss function defined in the methodology.

- Knowledge Distillation Models: Three novel applications of Knowledge Distillation models are used within the paper. Soft Target Distillation leverages softened output distributions from the teacher to guide the student. Feature Distillation using Probabilities introducing a regression head to align the student's probability vectors with those of the teacher. Feature Distillation using Embeddings aligning the student's internal hidden states with those of the teacher model.

4.5 Datasets

We evaluate all models across three datasets; the conflict dataset introduced in this paper (Breitsohl et al., 2018), (Table 1) with 4,052 texts across "Teasing", "Sarcasm", "Criticism", "Harassment", "Trolling", and "Threat" classes, a subset of the Davidson et al. dataset (Davidson et al., 2017) with texts across "Normal", "Offensive Language", and "Hate Speech" classes, and a subset of the Founta et al. dataset (Founta et al., 2018) with texts across "Abusive", "Hateful", and "Normal" classes. These datasets were selected to span a range of classification tasks both within the hate, negativity, and conflict domain and outside it, enabling evaluation of model generalization across varied task characteristics. Subsets were used for the Davidson et al. and the Founta et al. datasets as the original datasets contained very large numbers of datapoints which would have made training and evaluating models over these datasets prohibitively expensive for this study. We therefore chose to randomly sample subsets which preserve class balance. This allowed us to evaluate on these datasets to investigate model robustness whilst preserving computational efficiency. We reduce the size of each dataset to match the size of our conflict dataset with approximately 4,000 datapoints in each subset, equally distributed across all classes.

5 Experiments and Results

5.1 Model Performance

5.1.1 Experimental Setup

This experiment evaluates the performance of ConflictDT and knowledge distillation models against a set of state-of-the-art baselines. We conduct this evaluation across three datasets, our multi-class conflict dataset and two widely used hate speech datasets (Davidson et al. and Founta et al.), to assess generalization and robustness. Several researchers have emphasised the importance of cross-dataset generalisation in social media classification tasks (Fortuna et al., 2021; Barbiero et al., 2020), which is particularly relevant given the modest size of our primary dataset. By benchmarking ConflictDT across multiple domains, we demonstrate its adaptability and validate its comparative performance. This experiment also evaluates the Hierarchical model on the conflict dataset, enabling direct comparison with ConflictDT in a shared classification setting.

5.1.2 Results

Tables 3, 4, and 5 report classification model performance across three datasets (Conflict, Davidson, and Founta), using accuracy, F1, recall, and precision. The hierarchical model cells are empty for Davidson and Founta as only the conflict dataset had a hierarchical

Model	Acc	F-1	Rec	Pre
BERT	0.69	0.61	0.61	*0.65*
HateBERT	0.55	0.52	0.52	0.52
DistilBERT	0.55	0.53	0.52	0.54
GPT-2	0.59	0.51	0.51	0.55
Flan-T5	0.67	0.60	0.61	0.61
Conflict DT	*0.71*	*0.63**	*0.64*	0.62
Hierarchical BERT	0.69	0.62	0.63	0.63
KD Hidden Embedding	**0.72**	**0.67****	**0.67**	**0.67**
KD Logits Probabilities	*0.71*	0.61	0.63	0.62
KD Logits Combined Loss	0.70	0.60	0.61	0.61

Table 3: Model performance on Conflict dataset. Best results in bold, second best in italics. The notation ** and * denote statistical significance at p=0.05 and p=0.1 respectively.

Model	Acc	F-1	Rec	Pre
BERT	0.87	0.86	0.86	0.86
HateBERT	0.86	0.86	0.86	0.87
DistilBERT	0.85	0.86	0.86	0.86
GPT-2	*0.90*	0.73	0.72	0.73
Flan-T5	0.87	0.87	0.86	*0.88*
Conflict DT	0.89	*0.88**	*0.88*	*0.88*
Hierarchical BERT	-	-	-	-
KD Hidden Embedding	**0.91**	**0.90****	**0.90**	**0.90**
KD Logits Probabilities	0.86	0.85	0.85	0.85
KD Logits Combined Loss	0.86	0.85	0.85	0.85

Table 4: Model performance on Davidson et al. dataset. Best results in bold, second best in italics. The notation ** and * denote statistical significance at p=0.05 and p=0.1 respectively.

structure. The results allow us to answer R.Q.1 How effective are the different classification models for improving multi-class classification? How do the different forms of knowledge distillation compare in performance?

Overall, the best performing model was KD Hidden embeddings which consistently outperformed other models in all metrics. Its strong performance suggests that internal representation alignment is particularly effective for capturing nuanced behavioural distinctions, and may offer a more stable signal than output-level supervision. The ConflictDT model although significantly outperformed by KD Hidden Embeddings was the strongest among remaining models. This sets up the discussion in the performance cost analysis (section 5.5). We will now examine the results dataset by dataset to understand how different models compare in specific domains.

Model	Acc	F-1	Rec	Pre
BERT	0.77	0.76	0.77	*0.78*
HateBERT	*0.78*	0.73	0.73	0.73
DistilBERT	0.77	0.69	0.67	0.70
GPT-2	0.77	0.72	0.73	0.72
Flan-T5	0.77	0.76	0.76	*0.77*
Conflict DT	0.77	*0.77*	*0.78*	*0.78*
Hierarchical BERT	-	-	-	-
KD Hidden Embedding	**0.83**	**0.83****	**0.83**	**0.84**
KD Logits Probabilities	*0.78*	*0.77*	0.77	*0.78*
KD Logits Combined Loss	0.74	0.73	0.74	0.73

Table 5: Model performance on Founta et al. dataset. Best results in bold, second best in italics. The notation ** and * denote statistical significance at p=0.05 and p=0.1 respectively.

Within the conflict dataset, the best-performing model was the KD model with hidden state embeddings, which significantly outperformed other models. Whilst the KD Logit Probabilities model improves upon the baseline, it does not do so significantly. It showed modest gains, indicating that output-level supervision can be helpful but may lack the representational depth needed for subtle class boundaries. The KD Combined Loss model reduces performance compared to the baseline BERT model, suggesting that combining soft targets with classification loss—without access to raw logits—may dilute the learning signal or introduce conflicting gradients. These results highlight the importance of distillation design and underscore that not all KD strategies generalise equally across tasks. We also see that ConflictDT ranks second best, outperforming the baseline BERT model by 2% in F1 score and accuracy. This shows that the reward function within ConflictDT helps capture subtle differences. The HierBERT model results in a 1% increase in F1 score but no increase in accuracy; the increase in F1 score is not significant. Although this is not in line with the main experiment in the original HierBERT paper—which showed significant performance increases over the baseline BERT model—it does correlate with the experiments on the two additional tasks conducted in the paper. The additional tasks, like our paper, feature multiple subtasks in both higher-level groups, as opposed to just one in the main task of the HierBERT paper. The results suggest that the HierBERT model struggles when more complex sub-decisions must be made within the groups. Another reason for HierBERT's performance in this paper's task is that the difference between the two higher-level groups is less pronounced than in the original paper's main task. The difference between positive and negative emotions is more distinct than the differences between lesser and more extreme conflict behaviours—especially when some conflict behaviours straddle the border of the split. This idea will be explored more thoroughly in the next experiment, where we evaluate the ability of HierBERT and ConflictDT to differentiate between the two higher-level groups.

On the Davidson et al. dataset (Davidson et al., 2017), the KD hidden state embeddings model performed best whilst ConflictDT also outperformed all other variations of BERT by

2% in F1 score. However, it only marginally outperformed Flan-T5 by 1% in F1 score. GPT-2 outperforms ConflictDT by 1% in accuracy, although this is countered by significantly worse performance in F1 score, where ConflictDT outperforms GPT-2 by 15%.

As can be seen in Table 5, with the exception of the KD hidden state embeddings model, the ConflictDT model with a reward function of distance between all classes outperforms other models in F1 score on the Founta dataset (Founta et al., 2018). We achieve superior performance to BERT in F1 score by 1%, whilst also significantly outperforming GPT-2 by 5%. The best-performing model for accuracy was HateBERT with 0.78, although importantly, ConflictDT outperformed HateBERT in F1 score by 4% and matched all other models in accuracy at 0.77.

To test statistical significance, we performed paired t-tests between the ConflictDT model and the base BERT model. In each test, there were 4 degrees of freedom. For the conflict dataset, the t-value was 1.53 and the p-value was 0.08. For the Davidson dataset, the t-value was 1.62 and the p-value was 0.07. Therefore, we can say that our ConflictDT model significantly outperforms the base BERT model on the Davidson and Conflict datasets at a threshold of $\alpha = 0.10$, but not at $\alpha = 0.05$. For the Founta et al. dataset, the t-test did not show a significant p-value, with a t-value of 1.13 and a p-value of 0.15. Although this result was not significant using thresholds of either $\alpha = 0.05$ or $\alpha = 0.10$, the ConflictDT was still tested using 5-fold cross-validation and thus still shows merit in its performance.

Flan-T5 demonstrates strong generalisation across datasets, particularly in precision and recall, and performs competitively with ConflictDT and BERT variants. Its generative architecture may contribute to its robustness in handling diverse text styles. HateBERT, while achieving solid performance on the Founta and Davidson datasets, shows reduced performance on the conflict dataset—likely due to its pretraining on overtly abusive language, which may limit its sensitivity to subtler behaviours. GPT-2's high accuracy but low F1 score suggests overconfidence in dominant classes, resulting in poor balance across minority labels.

The results across three popular datasets show our ConflictDT model performs well within hate and conflict text classification. Although the best-performing model in the conflict dataset was the KD model with hidden embeddings, the ConflictDT model still ranks second best, outperforming the other two KD models. With regards to the F1 score metric, ConflictDT outperforms the other SOTA models on two datasets whilst achieving marginally better performance on the third. By showing that our model produces robust results, whilst also generalising across datasets, we increase the reliability of our results. This, therefore, mitigates the shortcomings of the conflict dataset, which suffers from class imbalance and data size. In addition, it also shows the adaptability of the reward functions for new data scenarios. Taken together, the comparative results suggest that KD Hidden Embeddings offers the best raw performance, Flan-T5 and HateBERT provide strong domain-specific baselines, and ConflictDT delivers a flexible, interpretable alternative with competitive results and lower computational cost. Finally, although KD models show robust performance, it does come at a higher computational cost. We will evaluate this performance-cost tradeoff later in the paper.

5.2 ConflictDT Reward Function Analysis and Sequential Modelling

5.2.1 Experimental Setup

This experiment investigates the role of reward functions and sequential modelling within the ConflictDT architecture, focusing on their impact on classification performance over the conflict dataset. Our approach in selecting individual reward functions involved a thorough analysis of distinct properties within the conflict dataset, specifically addressing existing misclassification patterns and class cross-talk.

We conducted a series of tests with multiple reward functions, examining the impact of each on overall classification performance. This setup showcases the novelty and flexibility of the reward functionality within ConflictDT, which allows the prioritization of different behaviours and aspects. To further analyse the performance of the BERT and ConflictDT models, we also include a breakdown of class-level performance within the conflict dataset.

The reward function was varied while keeping all other parameters constant. We tested five different reward functions:

i Distance between the text embedding and all classes

ii Distance between the text embedding and lesser and more extreme forms of conflict

iii Distance between the "Harassment" class and the text embedding

iv Distance between the text embedding and all classes with sequential functionality

These reward functions were selected based on our knowledge of the dataset and analysis conducted in section 3.5. The first reward function, "distance between all classes", aimed to prioritise separating all classes within the dataset and could be applied to any classification task. We propose that the extra contextual information will help the model reduce class cross-talk i.e. misclassification. The second reward function, "distance between lesser and extreme conflict groups", aimed to exploit common characteristics within these two groups, as identified during dataset analysis.

The "Harassment"-based reward function was derived from our model's initial results, which indicated that "Harassment" class datapoints were frequently misclassified into other classes and vice versa (see Fig.3). Consequently, we sought to demonstrate the model's ability to prioritise different classification behaviours by designing a reward function to address this trend.

The final two variations of ConflictDT feature sequential modelling. We show the impact of sequential modelling both with and without the reward functionality. This allows us to isolate the contribution of temporal structure and assess whether reward signals enhance or interfere with sequential learning.

We also compare the ability of ConflictDT (using the "lesser vs. extreme" reward function) and the Hierarchical model to differentiate between grouped conflict behaviours. This comparison helps evaluate whether explicit reward guidance or latent structural modelling is more effective in capturing behavioural distinctions.

	Acc	F-1	Rec	Pre
BERT	*0.69*	0.61	0.61	0.65
HierBERT	*0.69*	*0.62*	*0.63*	0.63
GPT-2	0.59	0.51	0.51	0.55
Flan-T5	0.67	0.60	0.61	0.61
Dist all Classes	**0.71**	**0.63**	**0.64**	0.62
Dist Less & Gtr Class Groups	0.67	0.59	0.62	**0.75**
Dist "Harassment"	0.68	0.61	*0.63*	0.67
Sequential no Reward	0.68	0.60	0.60	*0.74*
Sequential w/ Reward	*0.69*	*0.62*	0.62	0.64

Table 6: The effects of reward functions in the conflictDT model. Best results in bold, second best in italics.

5.2.2 Results

Table 6 shows the effects of different reward functions within the ConflictDT model, which produced a variety of classification performance scores. Here, we explore the potential of various reward functions in directing the classifier toward correct decisions. All reward-based variants of ConflictDT show comparable performance to the BERT baseline, with some outperforming it in specific metrics.

The ConflictDT model with a reward based on distances to all classes outperforms BERT by 4% in F1-score and 2% in accuracy. This variant achieved the best overall performance, demonstrating the effectiveness of general reward signals in guiding model decisions. As in section 5.1, BERT-based models outperform GPT-2 across all metrics.

Rewards based on the distance between lesser and more extreme conflict groups, and between "Harassment" and other classes, are slightly outperformed by BERT in overall accuracy. The "Harassment" reward matches BERT in F1-score, while the group-based reward sees a decrease of 0.02 in F1. Although these variants do not yield overall performance gains, they demonstrate the model's ability to influence behaviour at the class level.

Figure 8 shows individual class performance when using the Harassment reward. While overall metrics remain similar to BERT, the frequency at which other classes are misclassified as Harassment drops. Similarly, the group-based reward does not improve overall performance, but reduces the number of "Lesser" class datapoints misclassified into the "Greater" group. However, it also slightly increases the number of "Greater" datapoints misclassified as "Lesser", indicating a trade-off in directional control. These results show that ConflictDT can be used to influence model behaviour with respect to individual classes or structural properties of the classification task.

Examining BERT and ConflictDT classification heatmaps over the six-class conflict dataset (Figure 7), we observe that ConflictDT achieves higher performance in Teasing, Criticism, and Trolling; equal performance in Sarcasm; and lower performance in Harassment. Additionally, ConflictDT reduces the frequency of other classes being misclassified as Harassment, while Harassment is more frequently misclassified into Sarcasm and Trolling.

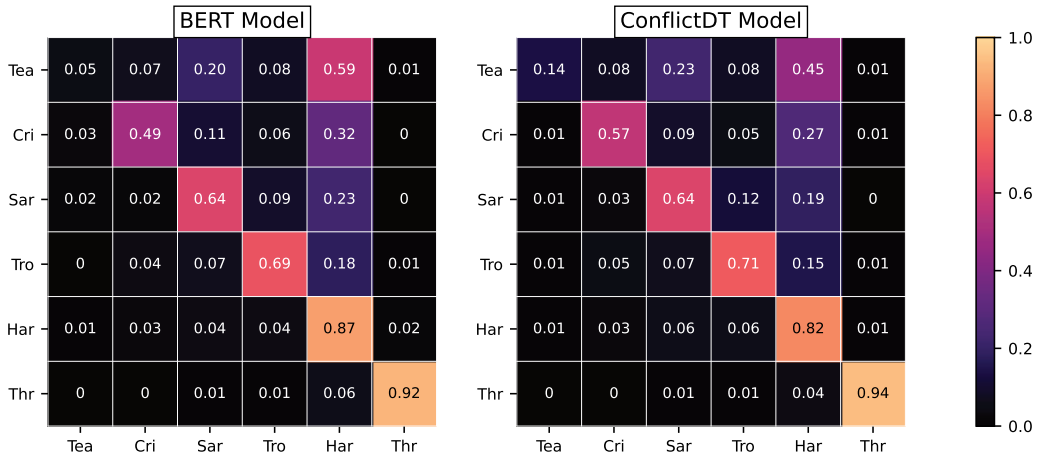


Figure 7: BERT and ConflictDT class performance on the conflict dataset.

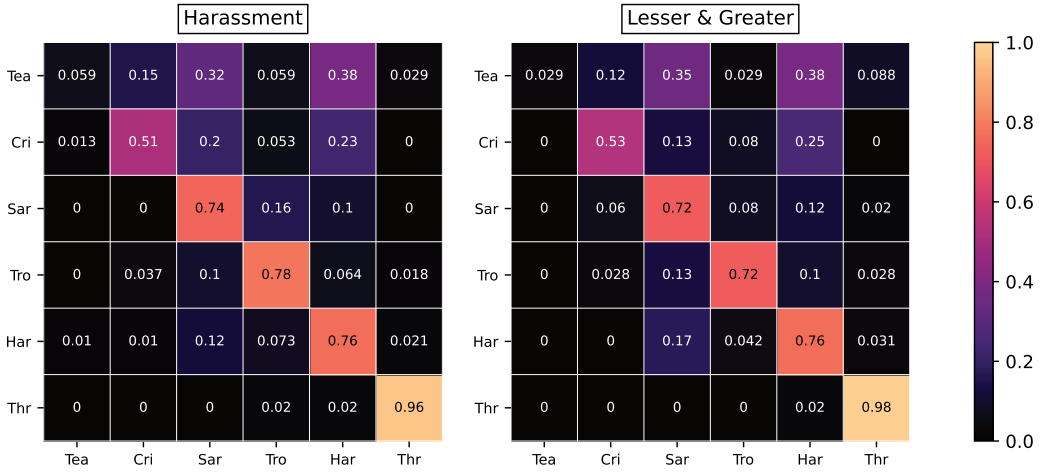


Figure 8: ConflictDT class performance with Harassment and Lesser-Greater rewards on the conflict dataset.

Considering the earlier dataset analysis in section 3.5, this outcome was unexpected. We had theorised that promoting distance between the two class groups would reduce misclassification between them. The observed decrease in performance may be due to ambiguity within each group or insufficient semantic separation between them.

Both sequential versions of the ConflictDT model fail to make significant improvements over the base model, with or without rewards. We theorise that this may be due to the short text length within the conflict dataset, quantified in Table 1. The additional data gained at each timestep may simply not be sufficient to aid the model, and may instead

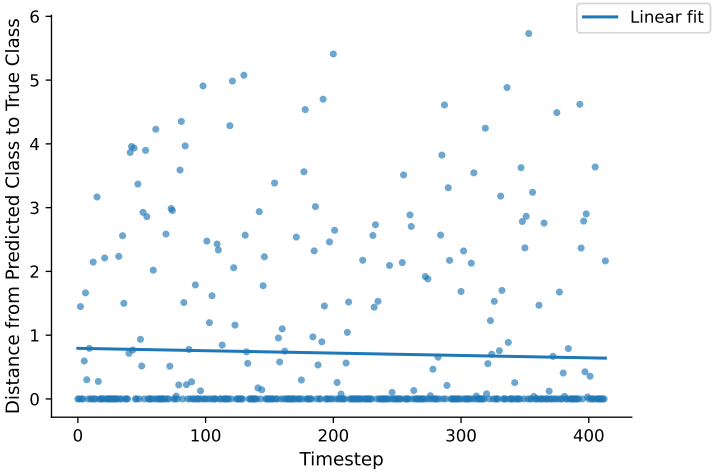


Figure 9: Graph showing the distance between the logits of the predicted class and the logits of the true class changing over timesteps.

introduce noise. To investigate this, we measured how the distance between the logits of the predicted class and the logits of the true class changed over time. If the sequential model worked as intended, this distance would converge toward zero as timesteps increase.

For all datapoints in the test set, we calculated the average distance between the predicted and true class logits at each timestep. Figure 9 shows that the average distance decreases slightly over time. We further calculated the Pearson correlation coefficient between distance and timestep, resulting in a coefficient of –0.05 and a p-value of 0.34. This suggests a weak negative correlation that is not statistically significant, indicating that sequential modelling does not meaningfully improve prediction in this context.

Model	Acc	F1	Rec	Pre
HierBERT	0.81	0.79	0.78	0.80
ConflictDT Less-Gtr	0.79	0.77	0.77	0.79

Table 7: Table evaluating the binary classification between the lesser and extreme conflict behaviour groups.

Table 7 presents binary classification performance between the lesser and extreme conflict behaviour groups. The HierBERT model outperforms ConflictDT with the group-based reward in both accuracy and F1-score. We can summarise that HierBERT is effective at distinguishing between the two groups, but ultimately does not improve classification at the second step when classifying into individual classes. This second decision is constrained by the initial group assignment—meaning that the 19% of datapoints misclassified at the first

step are automatically incorrect at the second. If the two conflict class groups were more distinct, methods that exploit these differences might be more successful.

In conclusion conflictDT with the reward of distance to all classes is the best performing variant. Targeted rewards such as harassment and lesser/greater distances show class level control but have limited increases in their overall performance. Sequential modelling shows no real gains, likely due to dataset characteristic constraints. A hierarchical model within the complex conflict class is effective at group separations but is limited by ambiguity within the groups. These findings highlight the flexibility of ConflictDT's reward design, whilst also providing insights into alternative models.

5.3 Empirical Analysis of Distillation Weight

5.3.1 Experimental Setup

This experiment evaluates the impact of distillation loss weighting on model performance across the three knowledge distillation strategies: Soft Target Distillation, Feature Distillation using Probabilities, and Feature Distillation using Hidden State Embeddings. We incrementally increased the weighting of the distillation loss relative to the classification loss, ranging from 1 to 10 in unit steps. This allows us to assess how sensitive each distillation method is to supervision strength and whether optimal weighting varies by strategy. As the raw distillation loss varies across models, using a uniform weighting scheme is unsuitable for direct comparison of absolute values. Instead, we focused on relative performance trends and identify optimal weighting ranges for each approach.

5.3.2 Results

The empirical analysis in Table 5.3.2 clearly shows that for each KD method, there exists an optimal distillation weight. For all three models, performance metrics decline both below and above this optimal point. For KD Logit Probabilities, the optimal distillation weight is 4.0; for KD Hidden Embeddings, it is 7.0; and for KD Soft Combined Loss, the optimal weight is 3.0, although 1.0 is a close second.

Interestingly, KD Hidden Embeddings is the best-performing model overall, KD Logit Probabilities ranks second, and KD Soft Combined Loss performs worst. This suggests a relationship between distillation weight and model effectiveness: more effective distillation strategies tend to benefit from higher weighting.

For the conflict classification task, the traditional classification loss was approximately 10 times larger than the distillation losses. Therefore, we explored a distillation weight range from 1.0 to 10.0. Had the two losses been of similar magnitude, a range from 0.1 to 1.0 may have been more appropriate. This highlights an important consideration: distillation weight—or any hyperparameter—is task-dependent, and fine-tuning is essential for optimal performance.

5.4 Thematic Analysis of Misclassifications

5.4.1 Experimental Setup

This experiment conducts a qualitative analysis of ConflictDT classifier outputs on the six-class conflict dataset, focusing on patterns of misclassification. To complement our

Wt.	Acc	F1	Pre	Rec	Acc	F1	Pre	Rec	Acc	F1	Pre	Rec
	KD Logit Probabilities				Feature Hidden States				Soft Distillation			
1	0.67	0.58	0.57	0.58	**0.72**	0.64	0.64	0.65	0.67	**0.60**	**0.61**	**0.61**
2	0.67	0.60	0.60	0.59	0.69	0.63	0.64	0.63	0.66	0.59	0.58	0.59
3	0.68	**0.61**	0.61	0.61	0.70	0.66	**0.70**	0.65	**0.70**	**0.60**	**0.61**	**0.61**
4	**0.71**	**0.61**	**0.62**	**0.63**	0.70	0.64	0.66	0.64	0.67	0.59	0.60	0.59
5	0.70	0.59	0.58	0.60	0.71	0.66	0.66	0.67	0.63	0.58	0.58	0.59
6	0.67	0.56	0.56	0.58	0.71	0.64	0.67	0.64	0.64	0.59	0.60	0.59
7	0.65	0.56	0.56	0.58	0.71	**0.67**	**0.67**	**0.68**	0.63	0.55	0.56	0.56
8	0.65	0.57	0.56	0.58	0.69	0.62	0.64	0.63	0.59	0.52	0.52	0.51
9	0.63	0.55	0.54	0.57	0.68	0.59	0.73	0.60	0.58	0.53	0.54	0.52
10	0.62	0.55	0.54	0.57	0.69	0.60	0.59	0.62	0.56	0.53	0.53	0.53

Table 8: Macro Performance Across Distillation Methods and Weights

quantitative evaluation, we apply thematic analysis—a widely recognised method in the social sciences for interpreting human communication—formally defined by Braun and Clarke (2006). Thematic analysis enables us to identify recurring themes and behavioural patterns within misclassified online comments. We follow the six-phase framework outlined by Braun and Clarke (2021): (1) Familiarization with the data, (2) Generation of initial codes, (3) Searching for themes, (4) Reviewing themes, (5) Defining and naming themes, and (6) Producing the report. One coder annotated the data, followed by a second coder who reviewed the identified codes, patterns, and themes. The two researchers then engaged in discussion to resolve discrepancies and finalise the thematic structure. The final analysis includes definitions, descriptions, and representative sample comments for each theme, enabling replication and interpretability. This approach provides insight into how model errors may reflect deeper communicative ambiguities or social dynamics.

5.4.2 Results

The thematic analysis produced three main misclassification themes, which we refer to as Linguistic Fluidity, Context Dependency, and Humour Ambiguity. These themes—alongside their definitions, descriptions, examples, and frequency—are presented in Table 9.

The theme of Linguistic Fluidity encompasses the blurred boundaries between class behaviours. Although datapoints tend to exhibit a dominant behaviour, they often contain aspects of multiple class behaviours. This ambiguity complicates both human and machine classification. For example, Jhaver et al. (2018) and Kim et al. (2022) identify crossover between Criticism and Harassment, discussing how Criticism can evolve into Harassment and how the perceived identity of a behaviour is often subjective. This theme also emerges in hate speech terminology, where Fortuna et al. (2020) highlight inconsistencies across research papers that lead to ambiguity between behaviour classes and misinterpretation of labels.

The second theme, Context Dependency, relates to datapoints where correct classification depends on conversational context that is unavailable to the model. While individual social media comments may contain overt negative behaviours, they are often embedded within larger threads or chains of interaction. Human annotators conducting netnographies

Theme	Definition	Description	Examples
Linguistic Fluidity	A misclassification that occurs due to the lack of definitional boundaries that are inherent to the interpretation of language.	A commonly known phenomenon in linguistics is that of multiple meanings to the same sentence, where meaning interpretation depends on a multitude of unpredictable factors(e.g. one's mood, need for politeness etc;) Classes are not always clear cut and often have fluid boundaries. Datapoints can contain behaviour which could belong to more than one class. Therefore neither classifier or human can get it totally accurate.	" Quickly! Let's spend our precious weekend time arguing about fast food on the Internet!" "everyone in every country has a right to their own opinion. I'm not sure why it matters what country, it's their opinions and their beliefs. No one is asking you to agree but respect the fact they are entitled to their opinions and beliefs."
Context Dependency	A misclassification that occurs when the meaning of a message changes when taken out of context.	Context dependency is a common issue identified by language researchers when studying offensive/impolite communication. The type of behaviour exhibited in a datapoint depends on the context of the prior comments. When classifiers are utilising single comment datapoints this context is not present, as opposed to the human coders who had access to the whole conversation when conducting netnographies.	"Picture that says: it's a joke not a dick, don't take it so hard." "If Andy Dwyer wasn't likable."
Humour Ambiguity	A misclassification that occurs when a message fails to convey that it was meant in humour and/or was good vs bad-natured.	Linguists have long recognised the lack of clarity inherent to humour as a quality on which humour often relies. Humour has been recognised as a particularly challenging area of NLP. Humour can be taken two ways and is subjective meaning the dominant type of humour behaviour is often ambiguous within datapoints.	"The Not So Fresh Princess of Stale Air." "Keep checking on Arsenal every champions league game mate ! Mu won't be there! Arsenal are here to entertain! unlike your confused club!"

Table 9: Table showing the themes identified in one test set's misclassifications

Model	Runtime (seconds)	Carbon Footprint (g CO2e)	Energy Consumption (Wh)
ConflictDT	799.60	2.19	23.01
Normal	761.92	2.06	21.65
Sequential	2384.98	6.46	67.76
LLM Probs	8451.17	22.96	241.16
LLM Hidden	1115.31	3.03	31.84
LLM Soft	8445.70	22.95	240.99

Table 10: Table showing the computational cost of training classification models.

had access to full conversations, whereas classifiers typically operate on isolated datapoints. This lack of context leads to misclassification when the behaviour is contingent on prior or surrounding discourse. Many researchers have addressed this issue by developing models that incorporate conversational context into classification tasks (Qian et al., 2018; Dahiya et al., 2021; Sahnan et al., 2021; Chakraborty and Masud, 2022).

The final theme, Humour Ambiguity, reflects the difficulty NLP models face in identifying humour, particularly when it is subtle, culturally specific, or reliant on shared knowledge. Humour is inherently subjective and often ambiguous, making it a persistent challenge in NLP. For example, the first humour ambiguity datapoint in Table 9, "The Not So Fresh Princess of Stale Air," was labelled as "Trolling" but misclassified as "Harassment". The model failed to detect the humourous intent and cultural reference to a 1990s sitcom, which a human reader would likely interpret as playful rather than hostile. The second example illustrates how teasing between football fans relies on shared knowledge of club rivalries and fan culture. Without this contextual grounding, the model struggles to distinguish banter from aggression.

5.5 Performance-Cost Analysis

5.5.1 Experimental Setup

This experiment evaluates the trade-off between classification performance and training cost across six models: ConflictDT, Sequential ConflictDT, KD Embedding, KD Probabilities, KD Soft Loss, and a baseline BERT model. We record the time taken to execute training loops for each model and use this as a proxy for computational expense. To extend this analysis beyond runtime, we apply the Green Algorithm's carbon cost estimator introduced by Lannelongue et al. (2021). This tool estimates the carbon footprint of any computational task by taking into account GPU type and count, memory usage, runtime, and platform. Using this framework, we present a novel accounting of the environmental cost of training classification models, enabling comparison not only in terms of accuracy but also environmental impact.

5.5.2 Results

The results shown in Table 10 illustrate the computational costs of the six evaluated models. Comparing ConflictDT with the BERT model, we observe slight increases of 4.95% in

runtime, 6.31% in carbon footprint, and 6.28% in energy consumption. The Sequential model, as expected, incurs significantly higher training costs, with increases of 314% in runtime, 313% in carbon footprint, and 298% in energy consumption.

The introduction of LLM-based KD methods results in a substantial increase in computational cost. Both KD Logit Probabilities and KD Soft Combined Loss models exhibit runtimes and carbon footprints far exceeding all other models. Given their relatively modest performance gains, these methods appear unsuitable for the conflict classification task. KD Hidden Embeddings also incurs higher runtime, but the increase is more modest and presents a realistic option.

Evaluating performance against computational cost, we find that the Sequential models—despite costing roughly three times more to run—offer no improvement in classification performance. ConflictDT and KD Hidden Embeddings present a more nuanced trade-off. ConflictDT increases F1-score by 2% with only a 4.95% increase in runtime, while KD Hidden Embeddings improves F1-score by 6% but increases runtime by 46.4%. In other words, ConflictDT is more efficient in terms of performance gain per unit cost, whereas KD Hidden Embeddings delivers the highest raw performance.

In conclusion, the Normal classification model offers the best value per performance. However, in scenarios where performance is paramount, KD Hidden Embeddings should be preferred. If deploying a large LLM for distillation is not feasible, but improved performance is still required, ConflictDT offers a practical middle ground—delivering gains at only a modest increase in cost.

6 Discussion of Results

Here we present a summary of the key findings in our results in relation to the research questions defined at the start of this paper.

6.1 Effectiveness of Classification Models

We successfully answer RQ1. by showing that some classification models are highly effective and significantly outperform the baseline. Notably, knowledge distillation with hidden embeddings and ConflictDT were the top two models respectively. Not all knowledge distillation methods were equal in performance, with knowledge distillation with logits and combined loss underperforming the baseline. These increases in performance are vital for real world application, especially within a high stakes conflict classification scenario,

6.2 ConflictDT Reward Performance

We confirm RQ2. by showing that the reward function within ConflictDT can be used to influence model behaviour and increase classification performance. Not all proposed rewards increased performance, but even some of those still achieved some of their goals, for example "Lesser and Greater" reward decreases the number of datapoints misclassified between the groups even though it doesn't decrease the number misclassified overall significantly. This ability to influence model behaviour is highly promising and positions ConflictDT as a novel method within text classification.

6.3 Hierarchical Framework Performance

We were able to show that the hierarchical model outperformed the ConflictDT model which focused on the parent group but both models were successful at differentiating between the parent groups (Shown in Table 7). This answers RQ3, showing that both methods capture the latent structure within the dataset. The hierarchical model does so by strictly differentiating between the parent groups whilst ConflictDT does so via its more subtle reward based approach. This experiment and RQ3 reinforce ConflictDT's ability to exploit relationships between classes alongside individual class characteristics to improve performance.

6.4 Knowledge Distillation Hyperparameters

As mentioned we were able to show that knowledge distillation with hidden state embeddings and knowledge distillation with probability embeddings improved classification performance over the baseline. We were further able to show through an empirical study that distillation hyperparameters have a significant effect on model performance. Not all distillation methods had the same distillation weight for optimum performance indicating that tuning per method and task is required. Results also show diminishing returns when tuning hyperparameters too aggressively. These results provide a clear and satisfactory answer to RQ5.

6.5 Model Misclassification Patterns

This experiment resulted in some interesting findings through the interdisciplinary methods which answered RQ4. The thematic analysis revealed three themes of misclassification; linguistic fluidity, context dependency, and humour ambiguity. These model misclassification trends link to existing social science theory, showing that models mirror the difficulties experienced by humans when interpreting nuanced online conflict behaviours.

6.6 Model Performance-Cost Tradeoffs

This experiment succinctly answered RQ.6 by showing the performance cost tradeoff using three key examples. Both ConflictDT and KD improve over the baseline model but both require extra computation. ConflictDT improves performance over the baseline without requiring large teacher models, making it cost effective. KD, whilst clearly offering the best performance, requires expensive teacher training. This creates questions surrounding cost, efficiency, and scalability within real world deployment.

6.7 Implications for Moderation and Responsible AI

The findings of this study have implications for automated moderation systems deployed in online platforms. Most current moderation pipelines focus on detecting overt forms of abuse such as harassment or explicit hate speech. Our results demonstrate that models can also learn to distinguish between more subtle conflict behaviours, such as teasing, sarcasm, and criticism. Detecting these earlier-stage behaviours may enable platforms to identify potentially harmful interactions before they escalate into more severe hostility.

At the same time, the misclassification analysis highlights the challenges inherent in automating nuanced behavioural interpretation. Linguistic fluidity, humour, and context dependency mean that even advanced models struggle to interpret intent in some cases.

This reinforces the importance of deploying automated classifiers as assistive tools rather than fully autonomous moderation systems. Human oversight and contextual judgement remain essential components of responsible content moderation.

This work also contributes to ongoing interdisciplinary discussions between computer science and social science regarding the measurement and interpretation of online conflict behaviours. By combining machine learning evaluation with qualitative thematic analysis, the study demonstrates how computational models can both benefit from and contribute to broader theoretical understandings of online communication.

7 Threats to Validity and Limitations

7.1 Dataset and External Validity

While evaluation across three datasets improves the robustness of our findings, several limitations remain. The conflict dataset introduced in this work is relatively modest in size compared to many contemporary NLP benchmarks and exclusively originates within online brand community discussions. Although the Davidson et al. and Founta et al. datasets were included to assess generalisation, the results may not fully transfer to other social media platforms, domains, or communication settings. Furthermore, the datasets used in this study are primarily English-language datasets and therefore do not capture linguistic and cultural variation that may influence the expression and interpretation of conflict behaviours. Future work should evaluate the proposed methods on larger, multilingual, and cross-platform datasets

7.2 Construct and Annotation Validity

The classification of online conflict behaviours is inherently subjective. Behaviours such as teasing, sarcasm, criticism, and harassment often exhibit overlapping characteristics and may be interpreted differently depending on cultural norms, conversational context, and annotator perspectives. Although the original dataset employed a dual-coding methodology to improve annotation reliability, some degree of ambiguity remains unavoidable. This limitation is further supported by the thematic analysis, which identified linguistic fluidity, context dependency, and humour ambiguity as recurring sources of misclassification. Consequently, the reported results should be interpreted as modelling socially constructed behavioural categories rather than objectively separable classes and any automated conflict detection should be interpreted cautiously and deployed with appropriate human oversight.

7.3 Experimental and Evaluation Validity

The study evaluates models using accuracy, precision, recall, and macro F1-score, which provide a broad view of classification performance across imbalanced classes. However, additional evaluation perspectives, such as calibration analysis, robustness testing, fairness assessment, and uncertainty estimation, were outside the scope of the present work. Furthermore, while the selected baselines represent widely adopted transformer architectures within text classification research, the evaluation does not include comparisons against more recent large instruction-tuned language models. Such comparisons would provide a stronger benchmark for assessing the relative effectiveness of the proposed approaches.

7.4 Reproducibility and Data Availability

To support reproducibility, all model architectures, training procedures, and hyperparameters are reported within the paper, and the implementations of ConflictDT, the hierarchical models, and the knowledge distillation approaches are publicly available. However, the conflict dataset contains sensitive social media content and cannot be distributed without restrictions. Access to the dataset is therefore provided on request for research purposes, subject to appropriate ethical approval and data-use considerations. While this may limit immediate replication, it represents a balance between research transparency and responsible handling of potentially harmful content.

8 Future Work

Future work should further evaluate the proposed approaches against stronger and more diverse baselines, including recent instruction-tuned and open-source large language models. Additional evaluation measures, such as calibration, robustness, uncertainty estimation, and fairness metrics, would provide a more comprehensive assessment of model behaviour in high-stakes moderation scenarios. Future studies should also investigate multilingual and cross-platform settings to better understand the generalisability of the proposed methods. To support continued research in this area, future work will focus on improving reproducibility through the public release of model implementations, training configurations, and evaluation pipelines, while maintaining controlled access to sensitive datasets.

Additionally, there is room for further exploration of reward functions. Our experiments focused on the distance between class text embeddings, but future reward functions could encompass other aspects, such as intra-class datapoint similarity, emotion and sentiment alignment, or behavioural clustering based on user intent. These alternatives may offer more targeted control over model behaviour and improve performance in ambiguous cases.

Research could also focus on utilising sequential modelling within text classification. The sequential variants of the ConflictDT architecture did not produce significant performance improvements in our experiments. One likely explanation is the short length of the social media posts within the datasets used in this study, which limits the amount of contextual information available for sequential modelling approaches to exploit. The Decision Transformer work by Chen et al. (2021) showed sequential modelling to achieve great success in other IR tasks. Given the sequential nature of text, the Decision Transformer framework could be exploited further, perhaps for long text forms or considering alternative representations of states within the text, such as individual words, different groups of words, or even paragraphs. Exploring hierarchical or multi-resolution state representations may help overcome the limitations observed in short-text scenarios.

Future work could also investigate more advanced knowledge distillation strategies tailored to conflict classification. For example, multi-teacher distillation could be used to transfer complementary behavioural insights from models trained on different domains (e.g. sarcasm detection, hate speech, trolling). Contrastive distillation techniques may help sharpen class boundaries by encouraging separation between semantically similar but behaviourally distinct classes. Additionally, task-adaptive distillation schedules—where the distillation weight varies dynamically based on class difficulty or model confidence—could offer more

flexible and context-aware supervision. Exploring multilingual distillation could also be valuable, particularly for cross-cultural conflict detection in global social media platforms.

Although we provide a qualitative analysis of our model's misclassifications using a popular and well-recognised social science technique, there is potential to perform a quantitative explainable AI (XAI) experiment. This would aid in understanding the role that the reward functionality plays within the model, and could help identify which features or signals are most influential in shaping predictions.

Finally, a case study investigating the use of this model within a simulated real-life scenario could be conducted, exploring its feasibility as a tool for use within social media platforms. Such a study would demonstrate the practical impact of a model capable of identifying a range of behaviours, while also highlighting responsible AI practices and ethical deployment considerations.

9 Conclusion

In this study, we introduce new modelling approaches for multi-class conflict classification and systematically compare three strategies—ConflictDT, hierarchical classification, and knowledge distillation—to assess their effectiveness. Our research findings indicate that the novel reward function within ConflictDT can successfully guide classification model behaviour and improve performance without relying on large teacher models. The hierarchical model was shown to be able to leverage the latent structure within the conflict groupings, although did not improve overall performance. Through the novel application of Knowledge distillation within the text conflict classification task we were able to replicate prior works within different domains to show that Knowledge distillation is a very effective method to improve classification performance. Together, this extensive study of classification models and techniques beyond baseline models advances conflict and text classification, highlighting the importance of analysing the full spectrum of conflict behaviours.

We also extend previous research by applying three knowledge distillation methods and evaluating distillation weights, showing that the performance of different knowledge distillation methods and hyperparameters are not uniform. To evaluate the effectiveness of the classification models we conducted a performance-cost evaluation, showing that although Knowledge distillation with hidden embeddings was the best performing model, ConflictDT achieved the most efficient performance increase. This underscores the importance of balancing performance with scalability. Finally, we conduct a qualitative analysis of model misclassifications, employing thematic analysis to identify trends and patterns within misclassifications. Themes such as linguistic fluidity, context dependency, and humour ambiguity reveal the limitations of current classifiers and highlight the need for socially informed modelling strategies. Overall, our research suggests the classification system and custom dataset introduced in this paper can serve as the foundation for further exploration into social media conflict.

Acknowledgments and Disclosure of Funding

Oliver Warke was supported by an EPSRC-funded PhD studentship at the School of Computing Science, University of Glasgow. No other additional external funding was received for this research.

References

Elias Aboujaoude, Matthew W Savage, Vladan Starcevic, and Wael O Salame. Cyberbullying: Review of an old problem gone viral. *Journal of adolescent health*, 57(1):10–18, 2015.

Karmanya Aggarwal, Pakhi Bamdev, Debanjan Mahata, Rajiv Ratn Shah, Ponnurangam Kumaraguru, et al. Trawling for trolling: A dataset. *arXiv:2008.00525*, 2020.

Waseem Akram and Rekesh Kumar. A study on positive and negative effects of social media on society. *International journal of computer sciences and engineering*, 5(10):351–354, 2017.

Davey Alba and Kurt Wagner. Elon Musk cuts more Twitter staff overseeing content moderation, Jan 2023. URL `https://www.bloomberg.com/news/articles/2023-01-07/elon-musk-cuts-more-twitter-staff-overseeing-content-moderation`.

Mohsan Ali, Mehdi Hassan, Kashif Kifayat, Jin Young Kim, Saqib Hakak, and Muhammad Khurram Khan. Social media content classification and community detection using deep learning and graph analytics. *Technological Forecasting and Social Change*, 188: 122252, 2023.

Fatimah Alkomah and Xiaogang Ma. A literature review of textual hate speech detection methods and datasets. *Information*, 13(6):273, 2022.

Brooke Auxier and Monica Anderson. Social media use in 2021. *Pew Research Center*, 1: 1–4, 2021.

Salvador V Balkus and Donghui Yan. Improving short text classification with augmented data using GPT-3. *Natural Language Engineering*, 30(5):943–972, 2024.

Pietro Barbiero, Giovanni Squillero, and Alberto Tonda. Modeling generalization in machine learning: A methodological and computational study. *arXiv:2006.15680*, 2020.

Chloe Berryman, Christopher J Ferguson, and Charles Negy. Social media use and mental health among young adults. *Psychiatric quarterly*, 89:307–314, 2018.

Federico Bianchi, Stefanie Hills, Patricia Rossini, Dirk Hovy, Rebekah Tromble, and Nava Tintarev. "It's not just hate": A multi-dimensional perspective on detecting harmful speech online. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pages 8093–8099, 2022.

Layla Boroon, Babak Abedin, and Eila Erfani. The dark side of using online social networks: a review of individuals' negative experiences. *Journal of Global Information Management (JGIM)*, 29(6):1–21, 2021.

Mondher Bouazizi and Tomoaki Ohtsuki. Multi-class sentiment analysis on Twitter: Classification performance and challenges. *Big Data Mining and Analytics*, 2(3):181–194, 2019.

Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

Virginia Braun and Victoria Clarke. Can I use TA? Should I use TA? Should I *not* use TA? Comparing reflexive thematic analysis and other pattern-based qualitative analytic approaches. *Counselling and psychotherapy research*, 21(1):37–47, 2021.

Jan Breitsohl, Holger Roschk, and Christina Feyertag. Consumer brand bullying behaviour in online communities of service firms. *Service Business Development: Band 2. Methoden–Erlösmodelle–Marketinginstrumente*, pages 289–312, 2018.

Thomas Brewster. Musk's X fired 80% of engineers working on trust and safety, australian government says, Aug 2024. URL `https://www.forbes.com/sites/thomasbrewster/2024/01/10/elon-musk-fired-80-per-cent-of-twitter-x-engineers-working-on-trust-and-safety/`.

Stefano Brogi. Online brand communities: a literature review. *Procedia-Social and Behavioral Sciences*, 109:385–389, 2014.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Taina Bucher, Anne Helmond, et al. The affordances of social media platforms. *The SAGE handbook of social media*, 1:233–253, 2018.

Matt Carlson. Embedded links, embedded meanings: Social media commentary and news sharing as mundane media criticism. *Journalism studies*, 17(7):915–924, 2016.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. HateBERT: Retraining BERT for abusive language detection in English. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25, 2021.

Tanmoy Chakraborty and Sarah Masud. Nipping in the bud: detection, diffusion and mitigation of hate speech on social media. *ACM SIGWEB Newsletter* (Winter):1–9, 2022.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

Justin Cheng, Cristian Danescu-Niculescu-Mizil, and Jure Leskovec. Antisocial behavior in online discussion communities. In *Proceedings of the international aaai conference on web and social media*, volume 9, pages 61–70, 2015.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53, 2024.

Snehil Dahiya, Shalini Sharma, Dhruv Sahnan, Vasu Goel, Emilie Chouzenoux, Víctor Elvira, Angshul Majumdar, Anil Bandhakavi, and Tanmoy Chakraborty. Would your tweet invoke hate on the fly? Forecasting hate intensity of reply threads on Twitter. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2732–2742, 2021.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 512–515, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

Ashwin Geet d'Sa, Irina Illina, and Dominique Fohr. Classification of hate speech using deep neural networks. *Revue d'Information Scientifique & Technique*, 25(01), 2020.

Clare Duffy. Meta gets rid of fact checkers and says it will reduce 'censorship', CNN business, Jan 2025. URL `https://edition.cnn.com/2025/01/07/tech/meta-censorship-moderation/index.html`.

Aleksandra Edwards and Jose Camacho-Collados. Language models for text classification: Is in-context learning enough? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10058–10072, 2024.

F Elsafoury. Cyberbullying datasets. *Mendeley*. URL `https://data.mendeley.com/datasets/jf4pzyvnpj/1`, 2020.

Hyung Chung et al. Hugging face, Google FLAN-T5 base, 2023. URL `https://huggingface.co/google/flan-t5-base`.

Paula Fortuna and Sérgio Nunes. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)*, 51(4):1–30, 2018.

Paula Fortuna, José Ferreira, Luiz Pires, Guilherme Routar, and Sérgio Nunes. Merging datasets for aggressive text identification. In *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 128–139, 2018.

Paula Fortuna, Juan Soler, and Leo Wanner. Toxic, hateful, offensive or abusive? What are we really classifying? An empirical analysis of hate speech datasets. In *Proceedings of the 12th language resources and evaluation conference*, pages 6786–6794, 2020.

Paula Fortuna, Juan Soler-Company, and Leo Wanner. How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? *Information Processing & Management*, 58(3):102524, 2021.

Antigoni Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. Large scale crowdsourcing and characterization of Twitter abusive behavior. In *Twelfth International AAAI Conference on Web and Social Media*, 2018.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A Smith. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. In *Findings of the association for computational linguistics: EMNLP 2020*, pages 3356–3369, 2020.

Nicolas Gontier, Pau Rodriguez, Issam Laradji, David Vazquez, and Christopher Pal. Language decision transformers with exponential tilt for interactive text environments. *arXiv:2302.05507*, 2023.

Reginald H Gonzales. Social media as a channel and its implications on cyber bullying. In *DLSU Research Congress*, pages 1–7, 2014.

Shloak Gupta, S Bolden, Jay Kachhadia, A Korsunska, and J Stromer-Galley. Polibert: Classifying political social media messages with bert. In *Social, cultural and behavioral modeling (SBP-BRIMS 2020) conference. Washington, DC*, 2020.

Georgios Hadjiharalambous, Kacper Beisert, and Joemon M Jose. End-to-end hierarchical approach for emotion detection in short texts. In *Responsible Data Science: Select Proceedings of ICDSE 2021*, pages 1–12. Springer, 2022.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv:1503.02531*, 2015.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Michael Janner, Qiyang Li, and Sergey Levine. Offline reinforcement learning as one big sequence modeling problem. *Advances in neural information processing systems*, 34:1273–1286, 2021.

Shagun Jhaver, Larry Chan, and Amy Bruckman. The view from the other side: The border between controversial speech and harassment on kotaku in action. *First Monday*, 23(2): 4, 2018.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. Tinybert: Distilling BERT for natural language understanding. In *Findings of the association for computational linguistics: EMNLP 2020*, pages 4163–4174, 2020.

Google Jigsaw. URL `https://www.perspectiveapi.com/#/home`.

Dacher Keltner, Lisa Capps, Ann M Kring, Randall C Young, and Erin A Heerey. Just teasing: a conceptual analysis and empirical review. *Psychological bulletin*, 127(2):229, 2001.

Muhammad US Khan, Assad Abbas, Attiqa Rehman, and Raheel Nawaz. Hateclassify: A service framework for hate speech identification on social media. *IEEE Internet Computing*, 25(1):40–49, 2020.

Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. A large self-annotated corpus for sarcasm. In *proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*, 2018.

Haesoo Kim, HaeEun Kim, Juho Kim, and Jeong-woo Jang. When does it become harassment? An investigation of online criticism and calling out in Twitter. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW2):1–32, 2022.

Taehyeon Kim, Jaehoon Oh, Nak Yil Kim, Sangwook Cho, and Se-Young Yun. Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 2628–2635. International Joint Conferences on Artificial Intelligence Organization, 2021.

Robin M Kowalski. "I was only kidding!": Victims' and perpetrators' perceptions of teasing. *Personality and Social Psychology Bulletin*, 26(2):231–241, 2000.

Robin M Kowalski, Gary W Giumetti, Amber N Schroeder, and Micah R Lattanner. Bullying in the digital age: a critical review and meta-analysis of cyberbullying research among youth. *Psychological bulletin*, 140(4):1073, 2014.

Robert V. Kozinets. The field behind the screen: Using netnography for marketing research in online communities. *Journal of Marketing Research*, 39(1):61–72, 2002. doi: 10.1509/jmkr.39.1.61.18935.

Robert V Kozinets. *Netnography: redefined*. Sage, 2015.

Mateusz Lango and Jerzy Stefanowski. What makes multi-class imbalanced problems difficult? an experimental study. *Expert Systems with Applications*, 199:116962, 2022.

Loïc Lannelongue, Jason Grealey, and Michael Inouye. Green algorithms: quantifying the carbon footprint of computation. *Advanced science*, 8(12):2100707, 2021.

Deborah Roth Ledley, Eric A Storch, Meredith E Coles, Richard G Heimberg, Jason Moser, and Erica A Bravata. The relationship between childhood teasing and later interpersonal functioning. *Journal of Psychopathology and Behavioral Assessment*, 28:33–40, 2006.

Kuang-Huei Lee, Ofir Nachum, Mengjiao Sherry Yang, Lisa Lee, Daniel Freeman, Sergio Guadarrama, Ian Fischer, Winnie Xu, Eric Jang, Henryk Michalewski, et al. Multi-game decision transformers. *Advances in Neural Information Processing Systems*, 35:27921–27936, 2022.

Jiacheng Li, Yujie Wang, and Julian McAuley. Time interval aware self-attention for sequential recommendation. In *Proceedings of the 13th international conference on web search and data mining*, pages 322–330, 2020.

Courtney Subramanian Liv McMahon, Zoe Kleinman. Meta to replace "biased" fact-checkers with moderation by users, Jan 2025. URL `https://www.bbc.co.uk/news/articles/cly74mpy8klo`.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The FLAN collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pages 22631–22648. PMLR, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv:1711.05101*, 2017.

Ariadna Matamoros-Fernández and Johan Farkas. Racism, hate speech, and social media: A systematic review and critique. *Television & New Media*, 22(2):205–224, 2021.

Meta and Kaplan, Joel. More speech and fewer mistakes. `https://about.fb.com/news/2025/01/meta-more-speech-fewer-mistakes/`, January 2025. Newsroom Post.

Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. Deep learning–based text classification: a comprehensive review. *ACM computing surveys (CSUR)*, 54(3):1–40, 2021.

Ioannis Mollas, Zoe Chrysopoulou, Stamatis Karlos, and Grigorios Tsoumakas. Ethos: a multi-label hate speech detection dataset. *Complex & Intelligent Systems*, 8(6):4663–4678, 2022.

Raymond T Mutanga, Nalindren Naicker, and Oludayo O Olugbara. Hate speech detection in Twitter using transformer methods. *International Journal of Advanced Computer Science and Applications*, 11(9), 2020.

Usman Naseem, Imran Razzak, and Ibrahim A Hameed. Deep context-aware embedding for abusive and hate speech detection on Twitter. *Aust. J. Intell. Inf. Process. Syst.*, 15 (3):69–76, 2019.

Esteban Ortiz-Ospina and Max Roser. The rise of social media. *Our world in data*, 2023.

Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv:1511.08458*, 2015.

Fabio Poletto, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, and Viviana Patti. Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation*, 55:477–523, 2021.

Jing Qian, Mai ElSherief, Elizabeth Belding, and William Yang Wang. Leveraging intra-user and inter-user representation learning for automated hate speech detection. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 118–123, 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Nils Reimers and Iryna Gurevych. all-mpnet-base-v2. `https://huggingface.co/sentence-transformers/all-mpnet-base-v2`, 2021. Accessed: 2025-11-14.

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets, 2015. URL `https://arxiv.org/abs/1412.6550`.

Dhruv Sahnan, Snehil Dahiya, Vasu Goel, Anil Bandhakavi, and Tanmoy Chakraborty. Better prevent than react: Deep stratified learning to predict hate intensity of Twitter reply chains. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 549–558. IEEE, 2021.

Joni Salminen, Maximilian Hopf, Shammur A Chowdhury, Soon-gyo Jung, Hind Almerekhi, and Bernard J Jansen. Developing an online hate classifier for multiple social media platforms. *Human-centric Computing and Information Sciences*, 10:1–34, 2020.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv:1910.01108*, 2019.

Vishwam Sankaran, Nov 2022. URL https://www.independent.co.uk/independentpremium/world/elon-musk-twitter-layoffs-moderators-b2224818.html.

Shabnoor Siddiqui, Tajinder Singh, et al. Social media its impact with positive and negative aspects. *International journal of computer applications technology and research*, 5(2):71–75, 2016.

Ge Song, Yunming Ye, Xiaolin Du, Xiaohui Huang, and Shifu Bie. Short text classification: a survey. *Journal of multimedia*, 9(5), 2014.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. BERT4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450, 2019a.

Siqi Sun, Yu Cheng, Zhe Gan, and Jingjing Liu. Patient knowledge distillation for bert model compression. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 4323–4332, 2019b.

Stuart A. Thompson and Kate Conger. Meet the next fact-checker, debunker and moderator: You, Jan 2025. URL https://www.nytimes.com/2025/01/07/technology/meta-facebook-content-moderation.html.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv:2302.13971*, 2023.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Well-read students learn better: On the importance of pre-training compact models, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Bertie Vidgen, Tristan Thrush, Zeerak Talat, and Douwe Kiela. Learning from the worst: Dynamically generated datasets to improve online hate detection. In *Proceedings of the 59th annual meeting of the Association for Computational Linguistics and the 11th international joint conference on natural language processing (volume 1: long papers)*, pages 1667–1682, 2021b.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. Do NLP models know numbers? Probing numeracy in embeddings. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5307–5315, 2019.

Jie Wang, Alexandros Karatzoglou, Ioannis Arapakis, Xin Xin, Xuri Ge, and Joemon M. Jose. Sparks of surprise: Multi-objective recommendations with hierarchical decision transformers for diversity, novelty, and serendipity. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, CIKM '24, page 2358–2368, 2024. Association for Computing Machinery. doi: 10.1145/3627673.3679533.

Jie Wang, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M. Jose, and Xuri Ge. Beyond accuracy: Decision transformers for reward-driven multi-objective recommendations. *IEEE Transactions on Knowledge and Data Engineering*, 37(9):5004–5016, 2025. doi: 10.1109/TKDE.2025.3582506.

Qiong Wang, Ruilin Tu, Yihe Jiang, Wei Hu, and Xiao Luo. Teasing and internet harassment among adolescents: The mediating role of envy and the moderating role of the Zhong-Yong thinking style. *International Journal of Environmental Research and Public Health*, 19(9):5501, 2022.

Tianyi Wang, Ke Lu, Kam Pui Chow, and Qing Zhu. Covid-19 sensing: negative sentiment analysis on social media in china via bert model. *Ieee Access*, 8:138162–138169, 2020a.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788, 2020b.

Xia Wang, Chunling Yu, and Yujie Wei. Social media peer communication and impacts on purchase intentions: A consumer socialization framework. *Journal of interactive marketing*, 26(4):198–208, 2012.

Oliver Warke, Joemon M Jose, Jan Breitsohl, and Jie Wang. Capturing the spectrum of social media conflict: A novel multi-objective classification model. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval*, pages 215–225, 2024.

Janis Wolak, Kimberly J Mitchell, and David Finkelhor. Does online harassment constitute bullying? an exploration of online harassment by known peers and online-only contacts. *Journal of adolescent health*, 41(6):S51–S58, 2007.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. Wikipedia Talk Labels: Personal Attacks. 2 2017a. doi: 10.6084/m9.figshare.4054689.v6. URL `https://figshare.com/articles/dataset/Wikipedia_Talk_Labels_Personal_Attacks/4054689`.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th international conference on world wide web*, pages 1391–1399, 2017b.

Lan Xia. Effects of companies' responses to consumer criticism in social media. *International Journal of Electronic Commerce*, 17(4):73–100, 2013.

Chuanpeng Yang, Yao Zhu, Wang Lu, Yidong Wang, Qian Chen, Chenlong Gao, Bingjie Yan, and Yiqiang Chen. Survey on knowledge distillation for large language models: methods, evaluation, and application. *ACM Transactions on Intelligent Systems and Technology*, 2024.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv:1612.03928*, 2016.

Mark Zuckerberg. More speech and fewer mistakes. Facebook Video, January 2025. URL `https://www.facebook.com/watch/?v=1525382954801931`.

Information Retrieval Research 2 (2026) 159-176 Submitted 1/26; Revised 06/26; Published 06/26

CASTlog: A Comprehensive Log of Children's Interactions with the Child Adaptive Search Tool

Christine Pinney CHRISTINEPINNEY@U.BOISESTATE.EDU
Boise State University
Boise, Idaho, USA

Casey Kennington CASEYKENNINGTON@U.BOISESTATE.EDU
Boise State University
Boise, Idaho, USA

Katherine Landau Wright KLWRIGHT@OREGON.EDU
University of Oregon
Eugene, Oregon, USA

Maria Soledad Pera M.S.PERA@TUDELFT.NL
TU Delft
Delft, Netherlands

Jerry Alan Fails JERRYFAILS@U.BOISESTATE.EDU
Boise State University
Boise, Idaho, USA

Editor: Johanne Trippas

Abstract

Children are increasingly interacting with search tools and exploring information online, whether it is for academic purposes or just for fun. Young searchers tend to favor search tools designed for adults and, as a result, face unique barriers to information access at all stages of the online search process. There have been many explorations into the obstacles children encounter during information retrieval (IR), but there is a lack of available data that can support further research from multiple perspectives and engage the larger IR community, resulting in stifled progress towards addressing the unique needs of young searchers. To address this lack of data, we introduce `CASTlog` a new dataset that captures the search queries of children ages 6-12 and the user-system interactions as searchers complete various kinds of online information discovery tasks. We also provide statistics of the data and an initial data analysis informed by empirical explorations based on the dataset. `CASTlog` serves as a foundational resource for researchers and practitioners focused on developing search tools that address the information needs of children.

Keywords: Dataset, Children Information Retrieval, Search, Query Analysis, Child-System Interaction

1 Introduction

As the digital information space continues to grow and technology becomes increasingly accessible, children are beginning to search online at younger and younger ages (Azpiazu et al., 2017; Anuyah et al., 2020; Bilal and Gwizdka, 2018). It is known that children largely

 WWW: https://irrj.org DOI: 10.54195/irrj.26178

prefer mainstream search engines (SE) like Google and Bing over those more specifically designed for them and their unique information needs (Azpiazu et al., 2017; Bilal and Huang, 2019; Danovitch, 2019; Gwizdka and Bilal, 2017), but mainstream SE have been designed and optimized for adults (Anuyah et al., 2020; Bilal, 2012; Bilal and Boehm, 2013; Bilal and Huang, 2019; Landoni et al., 2021), causing children to face many barriers when it comes to information access. Kuhlthau (1991) defines the *information seeking process* (ISP) as 6 sequential stages – initiation, selection, exploration, formulation, collection, and presentation – and outlines the thoughts, feelings, and actions of users that are characteristic of each stage. From initial query formulation to search result navigation and selection, children face distinctive obstacles within each of these stages (Pinney et al., 2024; Allen et al., 2021b; Bilal, 2012; Anuyah et al., 2020; Allen et al., 2022; Milton et al., 2020a; Allen et al., 2023; Collins-Thompson et al., 2011; Rajalakshmi et al., 2020; Azpiazu et al., 2017; Vanderschantz and Hinze, 2021; Gwizdka and Bilal, 2017). Ultimately, across different age ranges, children struggle to search for, retrieve, and utilize information from online resources (Bilal and Huang, 2019; Allen et al., 2022).

Recent research has explored the unique barriers to information access that children often face. In terms of query formulation, Pera et al. (2023) illustrate that children's unique strategies of query construction have significant implications on the resulting search results. Through analysis of logged queries gathered within the classroom context, the authors found that children tend towards natural language queries rather than keyword-based queries, where mainstream SE are better designed for the latter than the former (Pera et al., 2023). These findings reiterate earlier research carried out by Bilal and Gwizdka (2018), where children ages 11-13 were found to query using phrases and/or questions (70%) more often than keywords (30%). These authors also discovered that there are significant variations within this age range in terms of query formulation and reformulation strategies as well as levels of success in completing several types of search tasks, highlighting the need for "human intervention in terms of digital and search task literacy instruction, as well as system intervention to support children's query formulations and reformulations" (Bilal and Gwizdka, 2018, Section 9, p. 1038) during online search.

While these explorations (among many others not explicitly mentioned above (Landoni et al., 2022; Gwizdka and Bilal, 2017; Vanderschantz and Hinze, 2021; Landoni et al., 2019; Yarosh et al., 2018; Pera et al., 2023)) very clearly provide valuable insights regarding children's struggles with information retrieval (IR) and access, they are carried out via user studies with **limited users** (Landoni et al., 2019; Yarosh et al., 2018; Pinney et al., 2024; Gossen et al., 2014; Druin et al., 2010; Fails et al., 2019) or empirical explorations with **big but non-public data samples** (Landoni et al., 2020b, 2022; Bilal and Gwizdka, 2018; Gwizdka and Bilal, 2017; Vanderschantz and Hinze, 2021; Downs et al., 2020b; Anuyah et al., 2020), **non-contextualized data samples** without interaction (Pera et al., 2023), and **synthetic data samples** (Fröbe et al., 2025; Anuyah et al., 2020). A common denominator hindering the advancement of knowledge in terms of better supporting children's information access is the lack of available data that can enable research from multiple perspectives and comparison across solutions (Pera et al., 2025). Data and benchmarks from real children performing information discovery tasks using SE are scarce. This scarcity stifles progress in addressing challenges encountered by this often overlooked user group – namely, young searchers between the ages of 6 to 12 years old – resulting in a lack of engagement by

the wider IR research community and missed opportunities to enable the development of better IR technologies for children (Pera et al., 2025). Children of this age group are of particular interest as they are emergent searchers actively learning to read and develop their vocabularies and decoding skills as well as learning to search and develop their digital literacy (Kennington et al., 2021).

To address this gap, we introduce `CASTlog`[1], a new dataset constructed from logged data collected as a result of children conducting open-Internet searches, as well as targeted information discovery tasks related to specific lessons aligning with the primary school curriculum. More specifically, `CASTlog` is comprised of 13,309 unique queries from 7,931 unique search sessions carried out by children between the ages of 6 and 12 using the Child Adaptive Search Tool (CAST)[2], a search tool tailored to younger searchers. Co-designed with and for children, CAST features a uniquely simplistic design interface with a child-oriented, rule-based, phonetic spellchecker that provides query suggestions better-aligned with children's linguistic development, empowering them to become better searchers (Downs et al., 2020a). The dataset is rich with detailed information on queries, retrieved results, event data, and user interactions from live search sessions during which children progress through the various stages of online search.

Existing data sources related to children's search behavior are limited in both scope and accessibility. The dataset introduced in (Madrazo Azpiazu et al., 2018), while publicly available, includes only queries without interaction data. The datalog collected as a result of the study presented (Aliannejadi et al., 2021) contains queries and click data, yet remains private and inaccessible. The most recent resource (Fröbe et al., 2025) offers queries accompanied by labeled ground truth (determined by adults); however, it lacks authentic interaction data within an information-seeking process and does not reflect genuine child-generated queries, as these were formulated by adults on child-related topics. In contrast, `CASTlog` is distinctive in that it captures both authentic queries produced by children and their corresponding interaction behaviors. This combination of real queries and real user interactions sets it apart from existing datasets. Furthermore, search logs from commercial SE involving children have not been and are unlikely to ever be made publicly available due to strict policies and regulatory constraints. This further underscores the uniqueness of `CASTlog` particularly once it is released with appropriate ethical safeguards and necessary approvals in place. Overall, `CASTlog` constitutes a novel and valuable resource for research in Children's Information Retrieval, Child–Computer Interaction, and Digital Literacy. Its interactive nature, combined with its accessibility, fine-grained data structure, and educational context, provides a rare opportunity to examine the nuanced search behaviors of a protected and underrepresented user population—an opportunity that neither existing corporate nor academic datasets can adequately support.

In the rest of this paper, we discuss CAST and its role in the creation of `CASTlog`, outline the data collection process, describe how this ensemble dataset was produced, and highlight initial statistics on the collected data within `CASTlog`. We report on the outcome informed by empirical exploration based on this dataset, outline existing limitations, and paint a picture of the interplay between SE and children ages 6 to 12 across the ISP stages. To the best of our knowledge, `CASTlog` is the first of its kind and will be foundational for

1. `https://huggingface.co/datasets/BSU-CAST/CASTlog`
2. `https://cast.boisestate.edu/cast-simple/`

researchers and practitioners focused on research and developing algorithms and interfaces that respond to and address the needs of children when seeking information.

2 Child Adaptive Search Tool (CAST)

CAST is a child-centric search tool and user interface aimed at addressing the search and information needs of children ages 6-12 years old (i.e., emergent searchers). This search tool is part of a U.S. National Science Foundation-funded research project spearheaded by an interdisciplinary team and intended to empower young searchers by researching, designing, and developing search tools that improve children's information literacy and searching capabilities (see (Downs et al., 2019b; Allen et al., 2021b) for an overview).

The CAST search tool is powered by Microsoft Bing's SafeSearch. In addition to filtering unsafe and inappropriate search results, CAST presents a simplified and de-cluttered interface without advertisements to eliminate unnecessary distractions for younger searchers. CAST also employs a state-of-the-art spellchecker specifically designed to capture the unique misspellings of children and provide more effective support during query formulation when compared with mainstream SE designed with adults in mind (Downs et al., 2019a, 2020a,b). Although there have been other explorations with CAST in terms of re-ranking mechanisms and SERP design features, it is worth noting that these advances are not currently part of the tool (Allen et al., 2023; Pinney et al., 2024).

The CAST interface also includes a *lesson module*, where searchers are guided by educators via pre-recorded videos and prompts to complete curriculum-based information discovery search tasks. Lessons cover age and grade-appropriate topics (e.g., seasons and their characteristics, identifying elements on a map, and the structure and function of living things), and students' provided answers are logged.

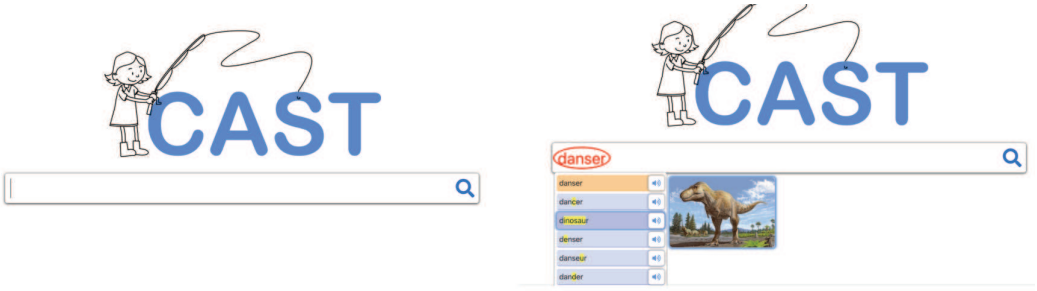


Figure 1: CAST main interface (left) and example spellchecker suggestion (right).

3 Data Collection

Data was collected using CAST over a period of 4 years, as a result of multiple explorations conducted to inform CAST design and deployment. These explorations focused on aspects such as query formulation/reformulation, query spellchecking, readability and ranking of retrieved results, SERP design and navigation, and evaluation of children's search tools. The specific protocols used varied across studies and are documented the associated prior

publications, which we enumerate as follows: Allen et al. (2023, 2021b, 2022); Milton et al. (2019); Pinney et al. (2024); Allen et al. (2021c); Garrett et al. (2021); Kennington et al. (2021); Allen et al. (2021a); Downs et al. (2020a,b); Landoni et al. (2020a); Milton et al. (2020b); Downs et al. (2019a); Fails et al. (2019); Murgia et al. (2019). Collected data was captured and logged from user-system interactions during unique search sessions using CAST.

While the ISP model has traditionally been developed and validated with adults and older students, our protocol design adapts its core stages (e.g., initiation, exploration, formulation, and collection) to align with the cognitive, developmental, and interactional characteristics of children aged 6–12. Different CAST modules were activated depending on the specific study. In controlled studies—encompassing both researcher-designed tasks and CAST lesson-based activities (see Section 4)—children engaged in a range of information-seeking tasks spanning entertainment and educational contexts, including targeted, exploratory, and open-ended search tasks, all using CAST. These studies were conducted in one of three environments: a lab setting (Boise State University), charter middle school classrooms (Boise, ID), or via Zoom (with U.S.-based participants). CAST also supported open Internet use outside formal study conditions, enabling free-search sessions not tied to specific experimental tasks; these interactions, limited to users aged 6–12, are also included in `CASTlog`. Further details regarding search tasks, participant demographics, and recruitment processes can be found in the previously cited research works. All collected data is in English.

4 Dataset

`CASTlog` is organized as tuples of search sessions and queries, further broken down into the individual **events** associated with each query, where events capture the interplay between users and the search system, and help illustrate the ISP process for individual queries within a search session. The dataset is organized in this manner to capture each individual query within a search session along with the many events (e.g., user-system interactions) that occur during ISP as a query is formulated and explored (i.e., keystrokes, query suggestions, retrieved results, and clicks). Figure 2 outlines the structure of the dataset and provides descriptions of each field. Each event type is associated with unique data contained within the *data* column. The *data* column is a key-value pair dictionary. The event types and contained data are:

- **query**: Each query event contains information about the query as well as the retrieved results.
  - *query*: the entered query string submitted to the search tool.
  - *query_len*: the length of the entered query in number of words.
  - *result_titles*: the titles of the top 10 results, presented as a list of strings.
  - *result_snippets*: the snippet of text that each of the top 10 results provide as a preview to result content, presented as a list of strings.
  - *result_urls*: the URLs of the top 10 results, presented as a list of strings.
  - *result_positions*: the position of each of the top 10 results within the presented ranked list of retrieved results, presented as a list of strings.

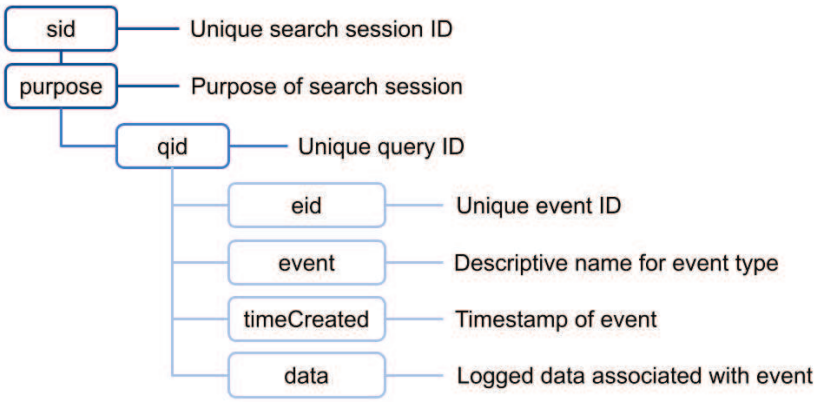


Figure 2: Overview of logged events, associated (meta)data, and their interconnection in `CASTlog`.

  - *result_SA_scores*: the Spache-Allen readability score of each of the top 10 result snippets, presented as a list of integers. The Spache-Allen readability formula, adapted from the original Spache readability formula presented in (Spache, 1953), assesses text difficulty using a calculation that analyzes sentence length and percentage of unfamiliar words (Allen et al., 2022). The formula produces a rounded integer score corresponding to the estimated reading grade-level of the text (i.e., a score of '2' implies a 2nd-grade reading level) and is particularly effective for children's reading materials (Allen et al., 2022). The formula is applied to each result's search snippet.

  Data about the retrieved results maintain order, meaning that the first list item in the *result_titles* list corresponds to the first list item of *result_snippets*, *result_urls*, *result_positions*, and *result_SA_scores* lists, and so on.

- **keystroke**: Each keystroke event contains data about keyboard keys pressed during query formulation.
  - *key*: the growing query string as each individual keyboard key is pressed during query formulation, presented as a string.
  - *pos*: the cursor position after a keyboard key is pressed, presented as an integer.
- **input-parsed**: Any input text in the search bar is periodically parsed for misspelled words.
  - *misspelled_words*: identified misspelled words within the growing query string, presented as a list of strings.
- **popup**: As misspelled words are identified, corresponding suggestion popup events provide spelling suggestions.

  - *misspell*: the misspelled word targeted for spelling suggestions, presented as a string.
  - *suggestions*: up to five suggested words based on the misspelled word, presented as a list of strings.

- **hov-suggestion** and **hov-off-suggestion**: Any hover mouse events are recorded to track mouse positions when spelling suggestions are presented during query formulation.
  - *hovered*: the position of the word within the spelling suggestion list that is hovered on or off, presented as a string.

- **hov-result** and **hov-off-result**: Any hover mouse events are recorded to track mouse positions during SERP navigation and exploration.
  - *hovered*: the position of the result within the result list that is hovered on or off, presented as a string.

- **clk-suggestion**: Any click mouse events are recorded to track mouse activity when spelling suggestions are presented.
  - *clicked*: the clicked spelling suggestion, presented as a string.
  - *position*: the position of the clicked spelling suggestion within the suggestion list, presented as a string.

- **clk-result**: Any click mouse events are recorded to track mouse activity during SERP navigation and exploration.
  - *position*: the position of the clicked result within the list of results, presented as a string.

- **lesson-start** and **lesson-next**: When the CAST lesson module is utilized, specific data is logged about the lesson and provided answers to prompts. *lesson-start* indicates when a lesson is started, and *lesson-next* indicates when the 'Next' button is pressed to move to a new question within a lesson.
  - *lsid*: a unique identifier for each lesson, similar to *sid* (the search session ID), presented as a string.
  - *answers*: a nested dictionary containing information about provided questions, question types, and entered responses. Each provided question within a lesson is stored as an individual key with an associated dictionary containing more specific information about the question as its value, including:
    * *id*: the question ID (e.g., *q1*, *q2*, etc.).
    * *type*: the question type (openanswer, mcq (multiple choice), or matching). Open answer questions specify a subtype: short or long. Multiple choice and matching questions provide an additional field listing the available choices.
    * *prompt*: the provided question/prompt.

    * *entered*: the entered response.

  Depending on the type of question and the specific lesson or topic, additional fields may be included.
- *topic*: the lesson topic, presented as a string. There two topics for 1st through 4th grade material, and one topic for 5th grade material.
- *grade*: the grade level of the lesson and topic, presented as an integer.

The dataset is annotated with pertinent metadata regarding exploration purposes to support and enable additional analyses and explorations from different perspectives; this metadata is contained within the 'purpose' column of the dataset. This metadata essentially allows for different 'slices' of the dataset to be easily accessed, providing ease of use within the IR community. Search sessions labeled as 'free search' contain all open-Internet data collected that is not associated with a particular lesson-centered search task and/or targeted research experiment. Search sessions labeled as 'research-driven search task' contain data associated with research-driven data collection, where queries and search result navigation are guided and/or directed by researchers for a specific purpose, but not associated with specific lessons. Search sessions labeled as 'research-driven lesson-based search task' contain all data associated with the outlined CAST lessons module and research-driven data collection. Search sessions labeled as 'self-directed lesson-based search task' contain all data associated with CAST lessons module that is not research-driven.

To ensure the dataset was constructed with consistent fields and data types, the logged data required some post-processing, including reshaping/reformatting and removing malformed data (i.e., missing data fields resulting from logging errors). Given the various ethical considerations across explorations, `CASTlog` does not include specific demographic information (age/grade, socioeconomic status, etc.) of users, and we recognize this as a limitation of the dataset.

Table 1 provides an initial overview of the data within `CASTlog` collected from online interactions with CAST during both open-ended use of the tool as well as controlled explorations, including the number of search sessions, unique queries, unique retrieved results, clicked results, and query suggestions. It is important to note that interaction events such as result clicks should not be treated as ground-truth indicators of the identification of relevant resources, successful information seeking, or learning outcomes. Instead, they merely reflect users' selection and navigation behaviors, and equating them with success may lead to misleading inferences from logged data. In the following section, we provide an initial analysis of the data within `CASTlog`.

5 Data Analysis and Future Explorations

Using the stages of ISP as a guide for analysis, we explore and illustrate some key observations emerging from `CASTlog`, discuss the implications for children's information access, and highlight opportunities for future explorations and analyses.

5.1 Initiation and Selection: Query Formulation and Suggestions

At the onset of ISP, searchers recognize the need for information (i.e., the initiation stage), leading them to select or identify a particular search task or topic for exploration (i.e., the

Overall	
Total # of Unique Search Sessions	7,931
Total # of Unique Queries	13,309
Total # of Unique URLs Retrieved	80,729
Total # of Query Suggestions Provided	60,547
Total # of Individual Events	1,632,480
Per Session	
Avg # of Click Events per Session	4.33
Avg # of Result Clicks per Session	3.84
Avg # of Queries per Session	3.55
Avg Session Length (in minutes)	8.87
Queries	
Total # of Non-Lesson Queries	17,308
Total # of Lesson Queries	5,644
Avg Query Length (in words)	3.86
Avg # of Stopwords in Query	1.46
Results	
Total # of Results Clicked	16,183
Avg Position of Results Clicked	2.85
Avg Spache-Allen Readability Score of Result Snippets	7.03
Spellchecker Suggestions	
Total # of Suggestion Hover Events	39,968
Total # of Suggestion Hover-Off Events	34,890
Total # of Suggestion Click Events	4,731
% of Total Suggestions Clicked	7.81
% of Total Suggestions Hovered	66.01
% of Total Suggestions Hovered-Off When Hovered	87.29

Table 1: Summarization of dataset statistics and user interactions.

selection stage) (Kuhlthau, 1991). Within these stages, searchers engage with the search system (i.e., CAST) and begin by formulating an initial query. `CASTlog` contains a total of 22,952 queries (13,309 unique queries), where 5,644 of those queries are related to the CAST lesson modules (i.e., those queries with the phrase 'lesson-based' included within the *purpose* column of the dataset). The average length of queries is 3.86 words, but Figure 3 displays some variation across this measure, revealing a tendency towards longer queries. These findings align with earlier research that reports longer queries resulting from children's query formulation (Duarte Torres et al., 2010; Madrazo Azpiazu et al., 2018). Queries contained an average of 1.46 *stopwords*, or commonly used words that carry little meaning and are often removed during SE optimization (e.g., *the*, *and*, *a/an*, etc.).[3] These findings imply that, on average, 1.5 words of a 4-word query are not keywords, resulting in queries following the pattern of natural language. This reiterates earlier findings that demonstrate a preference

3. Stopwords were identified using the `nltk.corpus.stopwords` module (Bird et al., 2009).

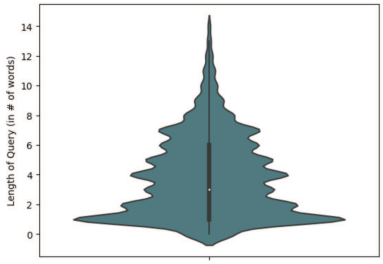


Figure 3: Distributions of query lengths across sessions.

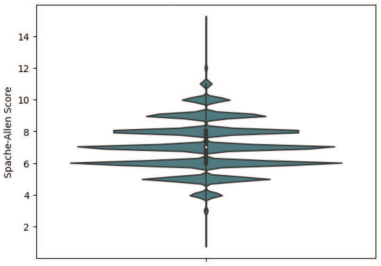


Figure 4: Distribution of Spache-Allen readability scores of retrieved results.

by younger searchers for natural language queries over keyword-based queries (Pera et al., 2023; Bilal and Gwizdka, 2018; Vanderschantz and Hinze, 2021).

`CASTlog` also contains information regarding user engagement with CAST's custom, state-of-the-art phonetic spellchecker tool that offers spelling suggestions while searchers formulate their queries (Downs et al., 2019a, 2020a,b). A total of 60,547 query spellchecking suggestions were provided to searchers during query formulation; of those suggestions provided, about 66% of the suggestions were hovered by users, but only 7.81% of the hovered suggestions were actually clicked on by users. Similar to findings presented by Druin et al. (2009) in their analysis of children's query formulation, these results reveal a lack of engagement with spelling suggestions during query formulation.

5.2 Exploration: Navigation of Search Results

After entering an initial query, searchers encounter a SERP and explore the information and resources provided (i.e., the exploration stage). `CASTlog` contains data about the results retrieved as well as the interactions carried out on the SERP by the searchers. The collected queries resulted in a total of 203,550 retrieved resources (80,729 unique URLs). To provide information on the readability of results, the dataset contains readability scores for each result that was retrieved for a given query. This calculation is applied to each result's search snippet, i.e., the text visible on the SERP extracted from the resource content (Bilal and Huang, 2019), using the Spache-Allen readability formula (Allen et al., 2022). This formula provides a rounded whole number estimating the reading grade level of the target text. On average, the Spache-Allen score for retrieved results is approximately 7, i.e., a 7th grade reading level; see Figure 4. This reiterates previous findings that report inaccessible readability of online resources for younger searchers (Bilal and Huang, 2019; Bilal, 2012; Collins-Thompson et al., 2011; Pinney et al., 2024). This has implications for children's ISP in particular, as unreadable or incomprehensible resources can be considerable barriers for children's information access. If a young searcher cannot read an online resource, they cannot extrapolate the information they are looking for and, thus, cannot complete ISP successfully. Mainstream SE are designed and optimized for adults, so the readability of results is not often taken into account by relevance algorithms employed by SE; this results

in SERPs with top-ranked results containing text that is too complex for younger searchers to comprehend (Bilal and Huang, 2019; Bilal, 2012; Collins-Thompson et al., 2011; Anuyah et al., 2020). Being a search tool tailored to the needs of younger searchers, CAST does employ a relevance algorithm that takes readability into account when ranking results on a SERP. While the returned results do include resources containing text at a more reasonable reading grade level for searchers (as shown in Figure 4), the youngest searchers of the 6-12 year old user group have fewer resources available, highlighting the larger issue of a lack of online resources that are readable for the youngest of online searchers.

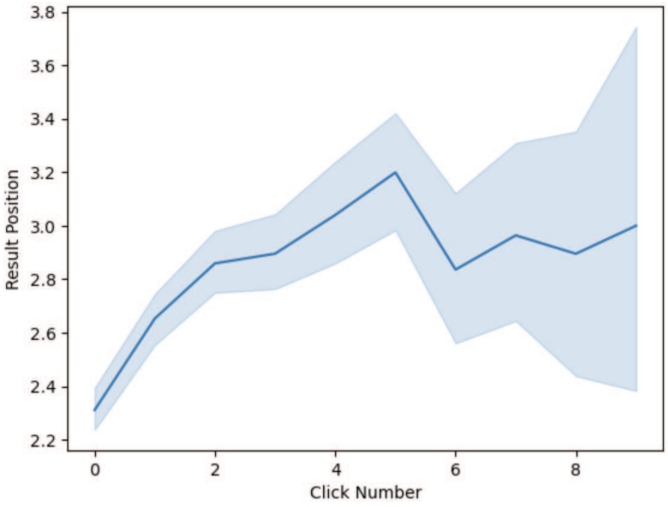


Figure 5: Average linearity of clicked results in sessions with multiple clicked results. Shading indicates the 95% confidence interval of the mean represented by the line.

In terms of user interactions on a SERP, `CASTlog` contains information about the position of clicked results within the retrieved list of results presented on a SERP. Data within `CASTlog` illustrate that a large majority of searchers displayed linear behavior (Landoni et al., 2022), i.e., often sequentially exploring ranked resources from the top result, in terms of clicked results (see Figure 5) and gravitated towards the top-ranked results on a SERP (see Figure 7). Linearity here refers to the relationship between the user's click order of results and the results' positions within the ranked list on the SERP. This reflects earlier findings that report considerable misconceptions by younger searchers about the significance and meaning behind ranking on a SERP, and a tendency towards exploring resources at the top of the presented list of results (Vanderschantz and Hinze, 2021; Bilal, 2012). As aforementioned, mainstream SE often have top-ranked results that are not readable for younger searchers, so their tendency towards exploring SERPs in a top-down, linear fashion increasingly complicates their experience with ISP. We caution researchers against explicitly equating clicked results with the success of a searcher in finding relevant answers, as this measure of success is not recorded in the dataset.

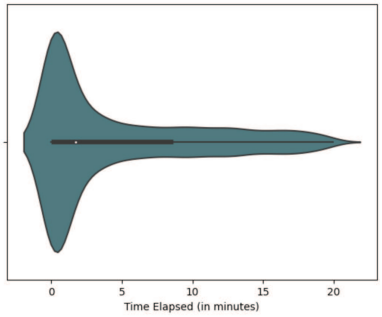


Figure 6: Distribution of time-elapsed during sessions.

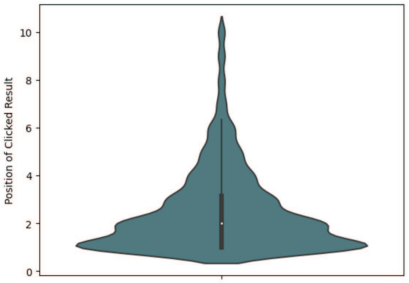


Figure 7: Distribution of positions of clicked results across sessions.

During the exploration stage of ISP, searchers investigate the presented information to extend personal understanding of the search topic, often encountering feelings of doubt and uncertainty, particularly when the information encountered does not align with previously held constructs and/or the searcher isn't able to effectively articulate what information is needed (Kuhlthau, 1991). With the unique barriers that young searchers face throughout ISP, this stage understandably becomes more difficult as they struggle to express search needs precisely during query formulation and successfully navigate the presented information. Some searchers are inclined to abandon the search altogether at this stage due to feelings of discouragement and frustration (Kuhlthau, 1991). This may result in shorter, more brief search sessions, as reflected by the decreased length of individual search sessions recorded within `CASTlog` (see Figure 6).

5.3 Dataset Usage and Ethical Considerations

There is a significant lack of publicly avialable query logs capturing the search behaviors of children in a holistic and ecologically valid manner. What currently exists, offers extremely limited support for empirical exploration and evaluation. In this context, `CASTlog` constitutes a rich, diverse resource capturing nuanced behavior along the key stages of the ISP, making it unique. The breadth and granularity of the logged interactions enable a wide range of research directions. `CASTlog` can be used to inform the understanding of children's information retrieval and the design and evaluation of IR technologies at the core of search (and other information access) systems. With that, we anticipate `CASTlog` supporting advancements related to several research directions that require attention when children are the main stakeholders, including strategies for query formulation and query suggestions, analysis of SERP results that draw attention from children and subsequent SERP navigation behaviors, patterns of interaction with information, and the design and assessment of (re-)ranking algorithms that support children's information discovery.

Moreover, the dataset provides a strong foundation for future extensions, such as the incorporation of ground truth aligned with child-centred notions of relevance, including

readability, educational alignment, and harm minimisation (Chakrabarti et al., 2026). This further strengthens its potential as a benchmark resource tailored to younger users.

While IR researchers may be the most apparent community that would benefit from this resource, there are additional communities that might find the data within `CASTlog` particularly valuable. For instance, the data related to the spellchecker and spellchecking suggestions during query formulation could offer rich insight for educators regarding spelling patterns and behaviors within this age group. The queries and results explored could also provide educators with insight into terminology and language comprehension. Additionally, data regarding mouse click behavior and hovering patterns could be incredibly useful to user interface and child-computer interaction researchers who aim to design and develop interfaces for younger audiences, providing insight into interface limitations and needs.

In light of the vulnerable user group at the center of this work, all ethical and legal considerations were carefully prioritized throughout the development of the dataset and in preparation for dissemination. We mindfully considered compliance with relevant protection laws, data privacy regulations (such as COPPA), and institutional review protocols. The dataset's purpose is strictly to support research advancements, particularly in IR, serving a population that has long been underrepresented and underserved (Pera et al., 2025, 2024). Moving forward, researchers and practitioners must remain aware of the ethical obligations associated with using and manipulating data from child users. They must proactively guard against misuse, assess potential risks, and ensure that any data used does not compromise the rights of this vulnerable population. Responsible use requires a balance between innovation and harm prevention, including scrutiny of research intentions, safeguards, and accountability mechanisms. Ultimately, `CASTlog` should be employed in ways that advance meaningful, positive outcomes for young users in a safe, respectful, and equitable manner.

6 Concluding Remarks

As children have increasingly been engaging with online search tools at younger and younger ages, it has become apparent that mainstream SE are designed with adult searchers in mind and fail to address the unique barriers that children encounter during ISP. The information needs of this particular user group are often overlooked by the IR community, and a lack of data contributes to this issue (Pera et al., 2025). `CASTlog` the first of its kind, is the result of a four-year multidisciplinary project. It serves as a foundation to address this gap and allow a wider array of research and perspectives to be explored to enable the design and development of better search tools for young searchers.

Acknowledgments and Disclosure of Funding

We thank the children who participated in the different studies that ultimately resulted in this data resource. We also appreciate the feedback from reviewers and editors, which helped improve the manuscript. This work was partially funded by NSF Award #1763649.

References

Mohammad Aliannejadi, Monica Landoni, Theo Huibers, Emiliana Murgia, and Maria Soledad Pera. Children's perspective on how emojis help them to recognise relevant results: Do actions speak louder than words? In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*, pages 301–305, 2021.

Garrett Allen, Brody Downs, Aprajita Shukla, Casey Kennington, Jerry Alan Fails, Katherine Landau Wright, and Maria Soledad Pera. BiGBERT: Classifying educational web resources for kindergarten-12th grades. In *Lecture Notes in Computer Science*, Lecture notes in computer science, pages 176–184. Springer International Publishing, Cham, 2021a.

Garrett Allen, Benjamin L Peterson, Dhanush Kumar Ratakonda, Mostofa Najmus Sakib, Jerry Alan Fails, Casey Kennington, Katherine Landau Wright, and Maria Soledad Pera. Engage!: Co-designing search engine result pages to foster interactions. In *Proceedings of the 20th Annual ACM Interaction Design and Children Conference*, IDC '21, pages 583–587, 2021b. Association for Computing Machinery.

Garrett Allen, Jie Yang, Maria Soledad Pera, and Ujwal Gadiraju. Using conversational artificial intelligence to support children's search in the classroom. *arXiv [cs.HC]*, 2021c.

Garrett Allen, Ashlee Milton, Katherine Landau Wright, Jerry Alan Fails, Casey Kennington, and Maria Soledad Pera. Supercalifragilisticexpialidocious: Why using the "right" readability formula in children's web search matters. In *European Conference on Information Retrieval*, pages 3–18. Springer, 2022.

Garrett Allen, Katherine Landau Wright, Jerry Alan Fails, Casey Kennington, and Maria Soledad Pera. Multi-perspective learning to rank to support children's information seeking in the classroom. In *2023 IEEE/WIC International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pages 311–317. IEEE, 2023.

Oghenemaro Anuyah, Ashlee Milton, Michael Green, and Maria Soledad Pera. An empirical analysis of search engines' response to web search queries associated with the classroom setting. *Aslib Journal of Information Management*, 72(1):88–111, 2020.

Ion Madrazo Azpiazu, Nevena Dragovic, Maria Soledad Pera, and Jerry Alan Fails. Online searching and learning: YUM and other search tools for children and teachers. *Information Retrieval Journal*, 20(5):524–545, 2017.

Dania Bilal. Ranking, relevance judgment, and precision of information retrieval on children's queries: Evaluation of Google, Yahoo!, Bing, Yahoo! kids, and Ask kids. *Journal of the American Society for Information Science and Technology*, 63(9):1879–1896, 2012.

Dania Bilal and A Reid Boehm. Towards new methodologies for assessing relevance of information retrieval from web search engines on children's queries. *Qualitative and Quantitative Methods in Libraries*, 2(1), 93–100, 2013.

Dania Bilal and Jacek Gwizdka. Children's query types and reformulations in Google search. *Inf. Process. Manag.*, 54(6):1022–1041, 2018.

Dania Bilal and Li-Min Huang. Readability and word complexity of SERPs snippets and web pages on children's search queries: Google vs bing. *Aslib Journal of Information Management*, 71(1), 2019.

Steven Bird, Edward Loper, and Ewan Klein. *Natural Language Processing with Python*. O'Reilly Media Inc., 2009. Stopwords Corpus compiled by Porter et al.

Hrishita Chakrabarti, Diletta Micol Tobia, Garrett Allen, Monica Landoni, and Maria Soledad Pera. It is relevant, but is it useful? A reflection on human-centred evaluation in children information retrieval. In *Proceedings of the 34th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '26, page 392–397, 2026. Association for Computing Machinery. ISBN 9798400723117. doi: 10.1145/3774935.3806167.

Kevyn Collins-Thompson, Paul N Bennett, Ryen W White, Sebastian De La Chica, and David Sontag. Personalizing web search results by reading level. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 403–412, 2011.

Judith H Danovitch. Growing up with Google: How children's understanding and use of internet-based devices relates to cognitive development. *Hum. Behav. Emerg. Technol.*, 1 (2):81–90, 2019.

Brody Downs, Tyler French, Katherine Landau Wright, Maria Soledad Pera, Casey Kennington, and Jerry Alan Fails. Searching for spellcheckers: What kids want, what kids need. In *Proceedings of the 18th ACM International Conference on Interaction Design and Children*, 2019a. ACM.

Brody Downs, Tyler French, Katherine Landau Wright, Maria Soledad Pera, Casey Kennington, Jerry Alan Fails, and Jerry Alan. Children and search tools: Evaluation remains unclear. In *Proceedings of KidRec '19*, 2019b.

Brody Downs, Oghenemaro Anuyah, Aprajita Shukla, Jerry Alan Fails, Sole Pera, Katherine Wright, and Casey Kennington. KidSpell: A child-oriented, rule-based, phonetic spellchecker. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6937–6946, 2020a.

Brody Downs, Aprajita Shukla, Mikey Krentz, Maria Soledad Pera, Katherine Landau Wright, Casey Kennington, and Jerry Fails. Guiding the selection of child spellchecker suggestions using audio and visual cues. In *Proceedings of the Interaction Design and Children Conference*, 2020b. ACM.

Allison Druin, Elizabeth Foss, Leshell Hatley, Evan Golub, Mona Leigh Guha, Jerry Fails, and Hilary Hutchinson. How children search the internet with keyword interfaces. In *Proceedings of the 8th International Conference on Interaction Design and Children*, 2009. ACM.

Allison Druin, Elizabeth Foss, Hilary Hutchinson, Evan Golub, and Leshell Hatley. Children's roles using keyword search interfaces at home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2010. ACM.

Sergio Duarte Torres, Djoerd Hiemstra, and Pavel Serdyukov. Query log analysis in the context of information retrieval for children. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, 2010. ACM.

Jerry Alan Fails, Maria Soledad Pera, Oghenemaro Anuyah, Casey Kennington, Katherine Landau Wright, and William Bigirimana. Query formulation assistance for kids: What is available, when to help & what kids want. In *Proceedings of the 18th ACM International Conference on Interaction Design and Children*, 2019. ACM.

Maik Fröbe, Sophie Charlotte Bartholly, and Matthias Hagen. How child-friendly is web search? An evaluation of relevance vs. harm. In *Lecture Notes in Computer Science*, pages 214–222. Springer Nature Switzerland, 2025.

Allen Garrett, Katherine Landau Wright, Jerry Alan Fails, Kennington Casey, and Maria Soledad Pera. CASTing a net: Supporting teachers with search technology. *arXiv [cs.CY]*, 2021.

Tatiana Gossen, Juliane Höbel, and Andreas Nürnberger. Usability and perception of young users and adults on targeted web search engines. In *Proceedings of the 5th Information Interaction in Context Symposium*, 2014. ACM.

Jacek Gwizdka and Dania Bilal. Analysis of children's queries and click behavior on ranked results and their thought processes in Google. In *Proceedings of the conference on conference human information interaction and retrieval*, pages 377–380. 2017.

Casey Kennington, Jerry Alan Fails, Katherine Landau Wright, and Maria Soledad Pera. Spellchecking for children in web search: A natural language interface case-study. In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 8–13, 2021.

Carol C Kuhlthau. Inside the search process: Information seeking from the user's perspective. *Journal of the American Society for Information Science*, 1991.

Monica Landoni, Davide Matteri, Emiliana Murgia, and Maria Soledad Pera. Sonny, cerca! evaluating the impact of using a vocal assistant to search at school. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction, Processings of the 10th International Conference of the CLEF Association, 2019*, pages 101–113. 2019.

Monica Landoni, Jerry Alan Fails, Theo Huibers, Natalia Kucirkova, Emiliana Murgia, and Maria Soledad Pera. 4th KidRec workshop - "what does good look like?": From design, research, and practice to policy. In *Proceedings of the ACM Interaction Design and Children Conference: Extended Abstracts*, 2020a. ACM.

Monica Landoni, Maria Soledad Pera, Emiliana Murgia, and Theo Huibers. Inside out: Exploring the emotional side of search engines in the classroom. In *Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization*, 2020b. ACM.

Monica Landoni, Mohammad Aliannejadi, Theo Huibers, Emiliana Murgia, and Maria Soledad Pera. Right way, right time: Towards a better comprehension of young

students' needs when looking for relevant search results. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '21, pages 256–261, 2021. Association for Computing Machinery.

Monica Landoni, Mohammad Aliannejadi, Theo Huibers, Emiliana Murgia, and Maria Soledad Pera. Have a clue! The effect of visual cues on children's search behavior in the classroom. In *Proceedings of the 2022 Conference on Human Information Interaction and Retrieval*, CHIIR '22, pages 310–314, 2022. Association for Computing Machinery.

Ion Madrazo Azpiazu, Nevena Dragovic, Oghenemaro Anuyah, and Maria Soledad Pera. Looking for the movie seven or sven from the movie frozen? a multi-perspective strategy for recommending queries for children. In *Proceedings of the 2018 conference on human information interaction & retrieval*, pages 92–101, 2018.

Ashlee Milton, Michael Green, Adam Keener, Joshua Ames, Michael D Ekstrand, and Maria Soledad Pera. StoryTime: eliciting preferences from children for book recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*, RecSys '19, pages 544–545, 2019. Association for Computing Machinery.

Ashlee Milton, Oghenemaro Anuya, Lawrence Spear, Katherine Landau Wright, and Maria Soledad Pera. A ranking strategy to promote resources supporting the classroom environment. In *2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pages 121–128. IEEE, 2020a.

Ashlee Milton, Levesson Batista, Garrett Allen, Siqi Gao, Yiu-Kai D Ng, and Maria Soledad Pera. Don't judge a book by its cover: Exploring book traits children favor. In *Fourteenth ACM Conference on Recommender Systems*, 2020b. ACM.

Emiliana Murgia, M Landoni, T Huibers, J A Fails, and M S Pera. The seven layers of complexity of recommender systems for children in educational contexts. *Proceedings of the Workshop on Recommendation in Complex Scenarios Co-Located with 13th ACM Conference on Recommender Systems*, Vol. 2449. CEUR-WS. pages 5–9, 2019. `http://ceur-ws.org/Vol-2449/paper1.pdf`

Maria Soledad Pera, Emiliana Murgia, Monica Landoni, Theo Huibers, and Mohammad Aliannejadi. Where a little change makes a big difference: A preliminary exploration of children's queries. In *Lecture Notes in Computer Science*, Lecture notes in computer science, pages 522–533. Springer Nature Switzerland, Cham, 2023.

Maria Soledad Pera, Emiliana Murgia, Monica Landoni, Federica Cena, Noemi Mauro, and Theo Huibers. Report on the 1st workshop on information retrieval for understudied users (IR4U2 2024) at ECIR 2024. In *ACM SIGIR Forum*, volume 58, pages 1–5. ACM, 2024.

Maria Soledad Pera, Theo Huibers, Emiliana Murgia, and Monica Landoni. Some things never change: Overcoming persistent challenges in children IR. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3107–3109, 2025. ACM.

Christine Pinney, Benjamin J Bettencourt, J A Fails, C Kennington, Katherine Landau Wright, and M S Pera. How readability cues affect children's navigation of search engine result pages. *Proceedings of the 23rd Annual ACM Interaction Design and Children Conference*, 62–69 2024. doi: 10.1145/3628516.3655818

R Rajalakshmi, Hans Tiwari, Jay Patel, R Rameshkannan, and R Karthik. Bidirectional gru-based attention model for kid-specific url classification. In *Deep Learning Techniques and Optimization Strategies in Big Data Analytics*, pages 78–90. IGI Global, 2020.

George Spache. A new readability formula for primary-grade reading materials. *The Elementary School Journal*, 53(7):410–413, 1953.

Nicholas Vanderschantz and Annika Hinze. Children's query formulation and search result exploration. *International Journal on Digital Libraries*, 22(4):385–410, 2021.

Svetlana Yarosh, Stryker Thompson, Kathleen Watson, Alice Chase, Ashwin Senthilkumar, Ye Yuan, and A. J. Bernheim Brush. Children asking questions: speech interface reformulations and personification preferences. In *Proceedings of the 17th ACM Conference on Interaction Design and Children*, IDC '18, page 300–312, 2018. Association for Computing Machinery. ISBN 9781450351522. doi: 10.1145/3202185.3202207.

Information Retrieval Research 2 (2026) 177-181 Published 6/2026

Editorial

Something to do with probability

Stephen Robertson STEPHENEROBERTSON@HOTMAIL.CO.UK
Cambridge, United Kingdom

I'm very pleased to write an editorial for IRRJ. It's great to see it into its second year of publication. It's a worthy successor to the *Information Retrieval Journal* that Paul Kantor and I started many years ago.

In the history of information retrieval, probabilistic models have a role to play, and in my research career, I have made use of them. The story I'm going to tell here has probability as one of its themes. That statement might make it sound like an abstract theoretical treatise, and I shall start that way, but actually its core is a very human story about the life and death of a colourful but flawed character who made a fortune as an entrepreneur, came near to losing it all, seemed to overcome potential disaster against the odds in 2024, and then in the same year died the most ridiculously unlikely death. I had a brief but interesting encounter with him in 2001.

Bayes

But let's start with the abstract bit. You may know about Bayesian statistics, but in case it's not familiar to you, here is a brief recap. Thomas Bayes was an eighteenth century Presbyterian minister, philosopher and statistician. In that last capacity, he developed a theorem in basic probability theory, a very useful theorem relating the probability of A given B to the probability of B given A.

A long time after Bayes, in the 20th century, a group of statisticians used Bayes' theorem as the basis for a rethinking of basic statistical ideas – the term Bayesian was not used until the 1950s. In the view of these statisticians (although there is more than one flavour), the probability of some event is not in general something that must be determined by a lot of experiments, like tossing a coin many times, but is a matter of something like a "degree of belief" – which should be modified in particular ways as evidence becomes available. Bayes' theorem is the basis for this modification. A "Bayesian" statistician is one who espouses this kind of philosophy of probability and statistics, in contrast to a "frequentist" statistician – this is commonly regarded as a major philosophical divide. But additionally, there has been a lot of very successful work developing Bayesian ideas in practical ways, to useful ends.

 WWW: https://irrj.org DOI: 10.54195/irrj.26692

Probabilistic IR

When I began working on a probabilistic model for IR in the early 1970s, the main precursor was the 1960 paper by Maron and Kuhns (1960). In this paper, they proposed the idea, and discussed possible methods, whereby a numerical probability could be assigned to the event of a user (having presented a query to the system) would find a particular document useful. In my work with Karen Spärck Jones, we used a similar argument: we proposed starting from some evidence about the probability that a relevant document would contain any particular word (if you like, the probability of a term given relevance). We then used Bayes' theorem to invert this, to estimate the probability of relevance given the presence of the term, and indeed to combine evidence across multiple query terms. The result was something like the Maron/Kuhns probability of relevance. I say 'something like' because there are some complications to our respective interpretations of "probability of relevance", but they need not concern us here.

Just for the record, this model (RSJ) was published exactly fifty years ago (Robertson and Spärck Jones, 1976), and was the precursor of BM25. I would not describe this model as "Bayesian" in any strong sense, despite the use of Bayes' theorem. Maron and Kuhns actually made rather more of the Bayesian aspect than we did.

After RSJ

After RSJ, in the late seventies, I had a joint project with Keith van Rijsbergen, based in Cambridge where Keith worked (I was at City University in London). Part of the project was to work on Harter's two-Poisson model of term frequency distributions (Harter, 1975). We did not have a lot of success with that model, though later it was to feed into BM25. But we employed as a researcher and software developer Martin Porter; he then designed and wrote the famous Porter stemmer (Porter, 1980), certainly the best known outcome of that project. After the project finished, Martin took the search software that he had developed, and formed a company with John Snyder, marketing the software as Muscat.

In the late 80s, I joined forces with Steve Walker, whose Okapi system was already running; it included relevance feedback with RSJ weighting. In the early 90s we took part in TREC, and under that competitive stimulus, we developed BM25.

I joined Microsoft Research Cambridge in 1998. The MSR Cambridge lab was run by Roger Needham who had previously directed the Cambridge University Computer Lab, effectively the University's computer science department. Also at the Computer Lab (but never employed at MSR) was Karen Spärck Jones, who was married to Roger, and with whom I had been working for over 20 years. I built a small team at MSR, and some of my work (particularly BM25) was incorporated into Microsoft's Sharepoint system, which was their product for corporate data management.

Mike Lynch

Now I must introduce to you the character I mentioned above: his name was Mike Lynch. (For those of you with long memories, there was another Mike Lynch active in the IR field – he worked at the University of Sheffield Information School from the 1960s on, and died at the age of 92, also in 2024 – but this is not him.) The Mike Lynch who concerns us here is the subject of an excellent 2025 biography by Katie Prescott, *The Curious Case of Mike Lynch* (Prescott, 2025). Apart from my own encounter with him in 2001, most of what follows is better and much more fully described, discussed and analysed in Katie's book.

Lynch was born in 1965, and got his PhD from Cambridge Engineering Department in 1990, in signal processing / pattern recognition. He is best known for founding Autonomy, in Cambridge in 1996, but before that he had a company called Cambridge Neurodynamics, which produced various kinds of software and hardware, including fingerprint recognition devices and character recognition. The company office was in the St. John's Innovation Centre, a haven for startups in Cambridge. Nextdoor was the Muscat office, with Martin Porter and John Snyder. Snyder and Lynch already knew each other.

Neither Lynch's research background nor the then work of Cambridge Neurodynamics included anything to do with text search or information retrieval; the Muscat software really interested him. He and Snyder started discussing a joint venture. However, Lynch decided to go his own way, and broke off with Snyder in 1994 – but not before Snyder had installed Muscat on one of the Cambridge Neurodynamics machines in order to demonstrate it to Lynch.

Then Lynch and Cambridge Neurodynamics developed their own text search and analysis software called the Dynamic Reasoning Engine, and in 1996 spun off a new company, Autonomy, to market the DRE.

Lynch was in the habit of making strong claims about the theoretical basis for the Autonomy software: he said, on every possible occasion, that it was based on Bayesian ideas. The whole of Autonomy's publicity, advertising and marketing, was aimed at convincing people that their Bayesian approach was utterly special and unique, a mantra, some kind of magical incantation. He stressed specifically that it was not simply a text search engine, and was not based on simple keywords, but that it could infer what documents were "really about".

Magic or no, over the next few years Autonomy became a very successful company, with (apparently) many corporate customers. It was floated on the stock exchange in 1998.

A fracas

I did not know Lynch or any of his associates, and I was barely aware of the company. But in early 2001, an analyst at the investment management company Merrill Lynch (another unrelated Lynch!) produced a report which suggested Autonomy stock was overvalued. Among other things, the report claimed that Microsoft was about to muscle in on Autonomy's market, and that Microsoft in Cambridge had poached important members of Lynch's company. It named – or rather misnamed – two people supposed to be working at MSR:

there was a slightly mangled version of Karen's name, and also one "Steve Robinson" (they could only have meant me). Autonomy's stock price promptly plummeted.

Internally, MSR Cambridge went into overdrive. Did they have *any* former Autonomy employees? (answer: no). Externally, it was mainly Mike Lynch who came out fighting – he laid a formal complaint against Merrill Lynch, which eventually forced them to withdraw parts of the report. Karen for her part spent a lot of effort trying, but in the end failing, to persuade Merrill Lynch to fund a studentship at the University Computer Lab as some kind of compensation.

Did I, or my group, or MSR Cambridge ever use any idea or method of Lynch's? I don't think so – Lynch never published his methods, and I never had occasion even to use Autonomy software. Did Lynch ever use any ideas of mine, or indeed from the wider IR community? Very probably: my ideas up to the time I joined Microsoft were all both published and in the public domain, and of course there was an entire world of published IR research, including both TREC and SIGIR. He was absolutely free to use them, though he never would have admitted to doing so.

But probably more important than the public status of these ideas was the fact that Lynch had access to the Muscat software. Muscat was a cutting-edge system, with some of my ideas, but Martin Porter built it with full knowledge and understanding of that world. Just for example, the Porter stemmer itself was highly innovative, but Martin knew and understood its predecessor, the Lovins stemmer, which relied much more on a dictionary of suffixes. Whether or not Lynch used the Muscat code itself, he could see what it did, how it worked, and what it was capable of. It's hard to imagine that the DRE was not influenced by this knowledge.

What happened next

So much for my own encounter with Mike Lynch. Now, very briefly, the rest of his story.

Autonomy survived the Merrill Lynch incident, and carried on successfully, until in 2011 the company was sold to HP. Within a year, HP came to the conclusion that they had been conned – that the information they had been provided with about the client base and the commercial success of Autonomy was seriously inaccurate. What followed was a series of lawsuits against Lynch and some colleagues, both in the US and the UK.

A major civil lawsuit brought by HP in the UK went against Lynch, and his estate may yet be bankrupted as a result. But in 2024, in a criminal case in a US federal court in San Francisco, he and a colleague were acquitted on all charges (the former chief financial officer of Autonomy had earlier been convicted and imprisoned on related charges). After his acquittal, Lynch went on a celebratory cruise in the Mediterranean, on a yacht which he named *The Bayesian* – a huge yacht with a 72m mast (somewhat taller than the 15-storey building in which I live). In the course of this cruise, while moored at night off the coast of Sicily, the *Bayesian* was caught in a freak storm, capsized, and sank rapidly. Mike Lynch and several others were drowned. In a startling coincidence, the other colleague who had

been acquitted in the San Francisco trial was hit in a road traffic accident two days before the capsize, and died the day after.

The ramifications of the Autonomy debacle continue, even after Lynch's death. At the time of writing, a UK court has ruled on the damages that Lynch's estate must pay to HP, but that might yet go to appeal. Also the builder of the *Bayesian* has sued the company which owned it (in turn owned by Lynch's widow) for damages, alleging that the sinking was the fault of the crew and that nobody is buying their yachts any more as a result of the disaster.

In the end, his magic deserted him.

References

M.E. Maron and J.L. Kuhns. On relevance, probabilistic indexing and information retrieval. *Journal of the Association for Computing Machinery*, 7, 216-244, 1960.

S.E. Robertson and K. Spärck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27, 129-146, 1976.

S.P. Harter. A probabilistic approach to automatic keyword indexing (parts 1 and 2). *Journal of the American Society for Information Science*, 26, 197-206 and 280-289, 1975.

M.F. Porter. An algorithm for suffix stripping. *Program*, 14, 130-137, 1980.

K. Prescott. The Curious Case of Mike Lynch: The Improbable Life & Death of a Tech Billionaire. *Macmillan Business*, 2025.