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

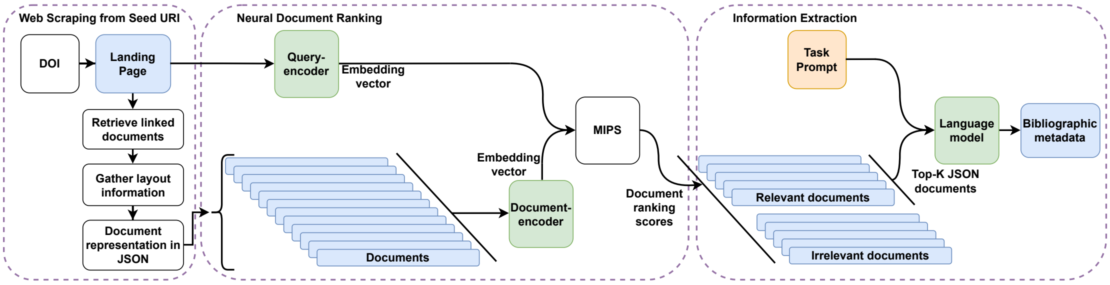


Figure 1: Illustration of the task of bibliographic information extraction from heterogeneous web sources. The process starts with an input DOI, which is resolved to access the associated landing page. We then employ a document-as-query approach to rank the documents linked from the landing page. A maximum 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 extraction. 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.

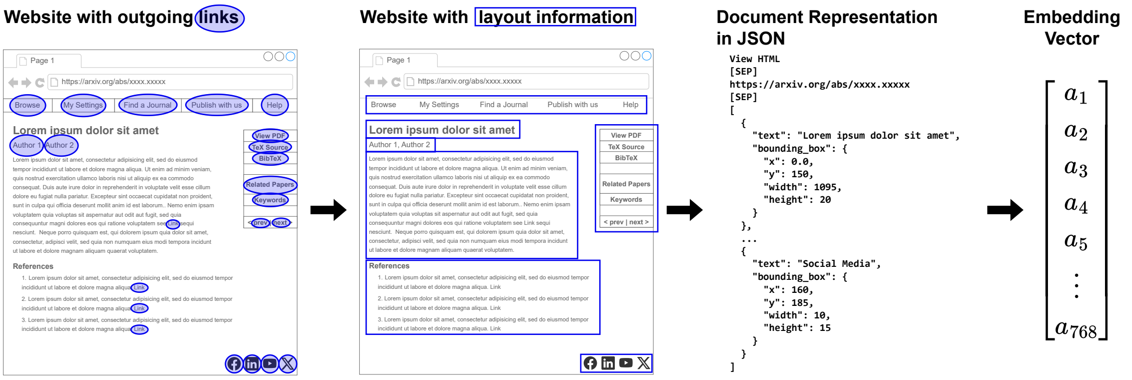
A seminal work on metadata collection and document-level metadata extraction is the CiteSeerX project (Li et al., 2006). Key to the system is a crawler for papers from authors' personal websites and other web sources. It employs a rule-based approach to detect affiliations, complemented by an SVM-based 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



```
View HTML
[SEP]
https://arxiv.org/abs/xxxx.xxxxx
[SEP]
[
  {
    "text": "Lorem ipsum dolor sit amet",
    "bounding_box": {
      "x": 0.0,
      "y": 150,
      "width": 1095,
      "height": 20
    }
  },
  ...
  {
    "text": "Social Media",
    "bounding_box": {
      "x": 160,
      "y": 185,
      "width": 10,
      "height": 15
    }
  }
]
```

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

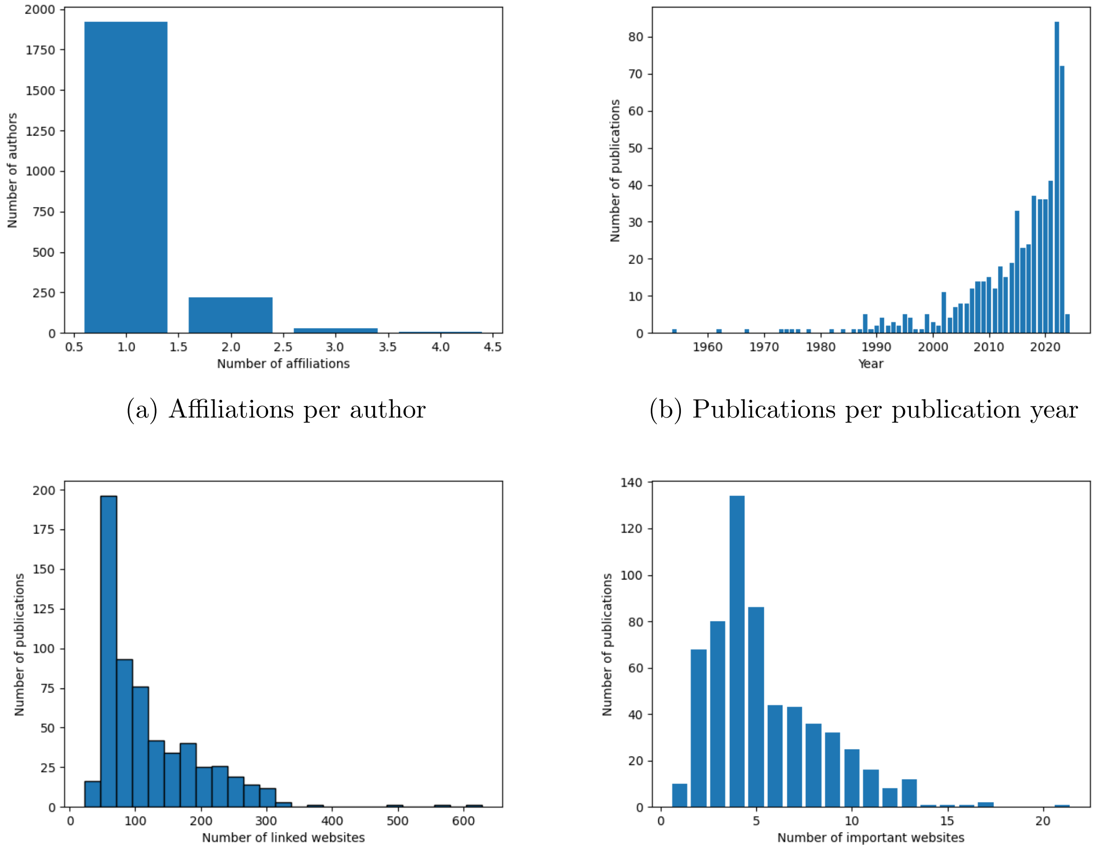
We use the DBLP Computer Science Bibliography dataset.[9] We take a subset of bibliographies from the six largest publishers in computer science, which together represent more than 80% of all publications listed in DBLP (see Appendix A for the full publisher distribution). This ensures the dataset contains 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

	Ranking			Extraction		
Publisher	**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

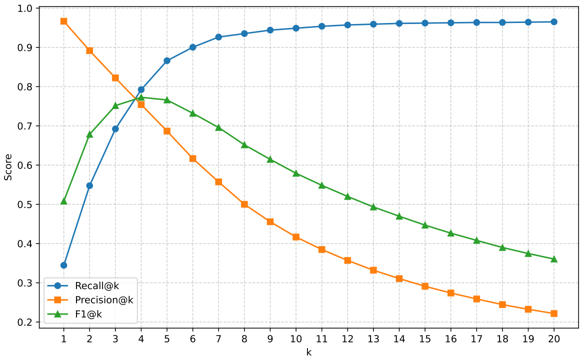


Figure 4: Visualization of the ranking performance evaluation of the model at different cut-off values k.

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

	Title			Author Names			Author 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
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.

	Names & Affiliations		
Publisher	**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.

	Title			Author Names			Author Affiliations			Publication Year		
Publisher	**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.

	Title			Author Names			Author Affiliations			Publication Year		
Publisher	**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-

	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	0.806	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.

	CRAWLDoc			Only the landing page		
Publisher	**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.

	Acc.	Acc.	DOI	
Count	**Count**	**Coverage**	**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.

	Jina-v2 (zero-shot)			BM25			BM25+			Jina-v2 (ours)		
Publisher	**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

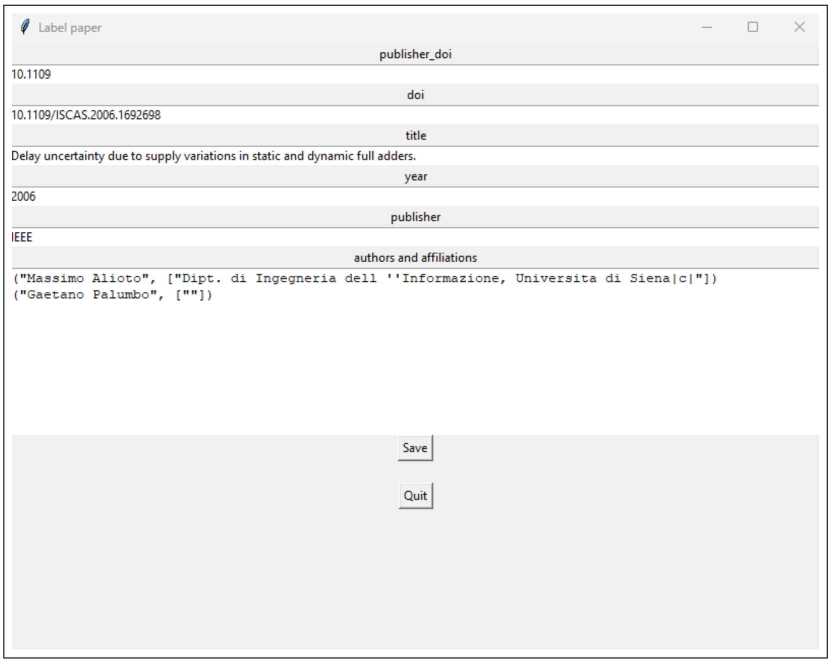
The process of labeling data for this task required a substantial amount of manual labor. To make the process more efficient, we created a specialized utility tool. Our tool employs a two-stage approach: In the first stage, we emulate a browser landing page and prompt the user for metadata input (as 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.



publisher_doi

10.1109

doi

10.1109/ISCAS.2006.1692698

title

Delay uncertainty due to supply variations in static and dynamic full adders.

year

2006

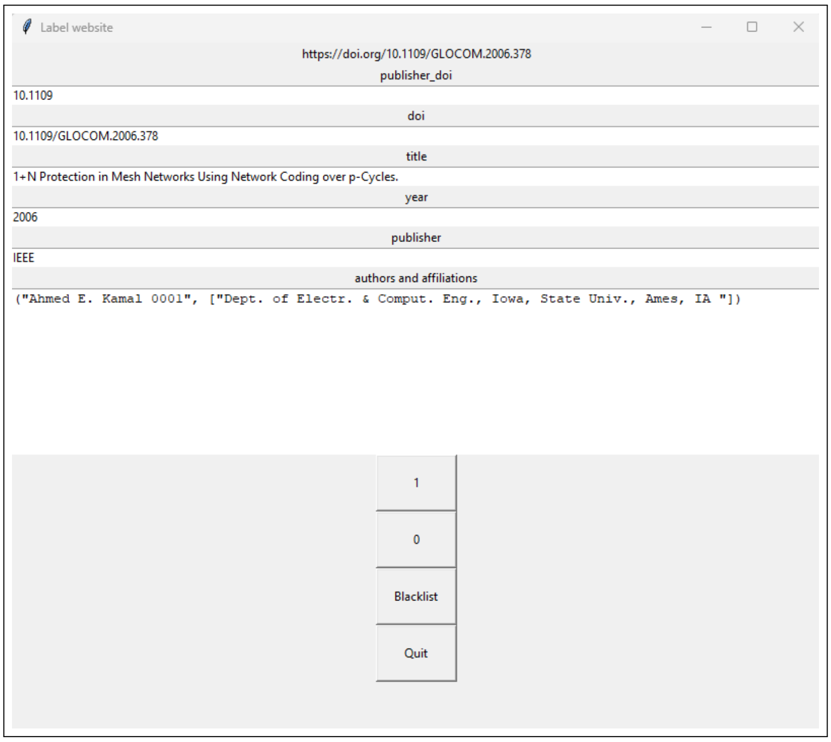
publisher

IEEE

authors and affiliations

```
("Massimo Alioto", ["Dipt. di Ingegneria dell ''Informazione, Universita di Siena|c|"])
("Gaetano Palumbo", [""])
```

Figure 5: The GUI of our metadata labeling tool



https://doi.org/10.1109/GLOCOM.2006.378

publisher_doi

10.1109

doi

10.1109/GLOCOM.2006.378

title

1+N Protection in Mesh Networks Using Network Coding over p-Cycles.

year

2006

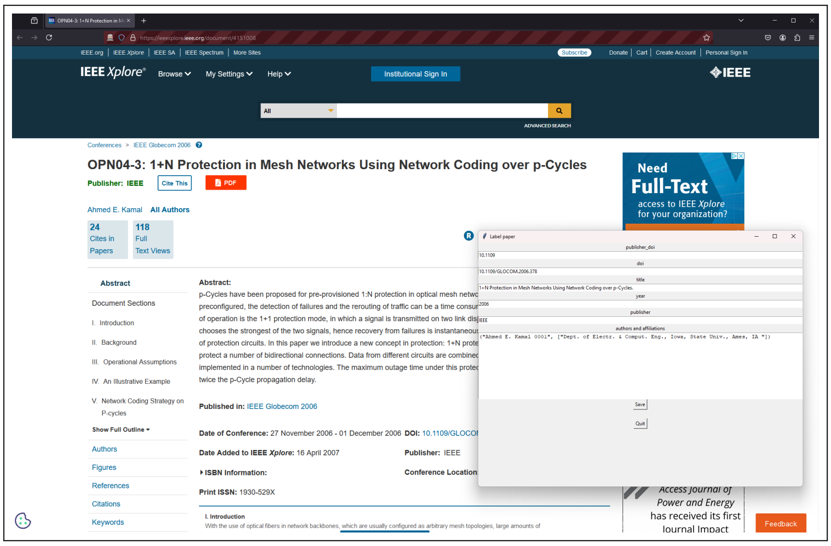
publisher

IEEE

authors and affiliations

```
("Ahmed E. Kamal 0001", ["Dept. of Electr. & Comput. Eng., Iowa, State Univ., Ames, IA "])
```

Figure 6: The GUI of our relevancy labeling tool


https://ieeexplore.ieee.org/document/4151008
IEEE.org | IEEE Xplore | IEEE SA | IEEE Spectrum | More Sites
Subscribe
Donate | Cart | Create Account | Personal Sign In
IEEE Xplore®
Browse
My Settings
Help
Institutional Sign In
IEEE
All
ADVANCED SEARCH
Conferences > IEEE Globecom 2006
Cite This
PDF
Need Full-Text access to IEEE Xplore for your organization?
Abstract
Document Sections
I. Introduction
II. Background
III. Operational Assumptions
IV. An Illustrative Example
V. Network Coding Strategy on P-cycles
Show Full Outline
Authors
Figures
References
Citations
Keywords
Save
Quit
Feedback

OPN04-3: 1+N Protection in Mesh Networks Using Network Coding over p-Cycles

Publisher: IEEE

Ahmed E. Kamal All Authors

24 Cites in Papers | 118 Full Text Views

Abstract:
p-Cycles have been proposed for pre-provisioned 1:N protection in optical mesh netwo
preconfigured, the detection of failures and the rerouting of traffic can be a time consur
of operation is the 1+1 protection mode, in which a signal is transmitted on two link disj
chooses the strongest of the two signals, hence recovery from failures is instantaneous
of protection circuits. In this paper we introduce a new concept in protection: 1+N prote
protect a number of bidirectional connections. Data from different circuits are combinec
implemented in a number of technologies. The maximum outage time under this protec
twice the p-Cycle propagation delay.

Published in: IEEE Globecom 2006

Date of Conference: 27 November 2006 - 01 December 2006 DOI: 10.1109/GLOCOI

Date Added to IEEE Xplore: 16 April 2007 Publisher: IEEE

ISBN Information: Conference Location:

Print ISSN: 1930-529X

I. Introduction
With the use of optical fibers in network backbones, which are usually configured as arbitrary mesh topologies, large amounts of

Label paper

publisher_doi: 10.1109

doi: 10.1109/GLOCOM.2006.378

title: 1+N Protection in Mesh Networks Using Network Coding over p-Cycles.

year: 2006

publisher: IEEE

authors and affiliations:

```
("Ahmed E. Kamal 0001", ["Dept. of Electr. & Comput. Eng., Iowa, State Univ., Ames, IA "])
```

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.