Information Retrieval Research 1 (2025) 281-312 Submitted 07/25; Revised 10/25; Published 12/25

Exploring Embedding Interpretability by Correspondences Between Topic Models and Text Embeddings

Meng Yuan MENG.YUAN@UNIMELB.EDU.AU
School of Computing and Information Systems
University of Melbourne, Melbourne, Australia

Lida Rashidi LIDA.RASHIDI@RMIT.EDU.AU
School of Computing Technologies
RMIT University, Melbourne, Australia

Justin Zobel JZOBEL@UNIMELB.EDU.AU
School of Computing and Information Systems
University of Melbourne, Melbourne, Australia

Editor: Solomon Atnafu

Abstract

Text embeddings have become essential for representing documents in Information Retrieval (IR), yet their high-dimensional nature often limits interpretability. To bridge this gap, we introduce a novel mapping framework that aligns embedding dimensions with topics derived from both probabilistic and neural models. Using three standard collections and three embedding methods, we demonstrate that embedding features consistently map to a subset of coherent topics, even as the total number of topics varies. We further quantify this correspondence with a Mean Mapping Specificity Improvement Rate, showing that mapped topics exhibit significantly higher specificity than the global topic set if the embedding dimensions are set properly. A stability analysis over varying embedding dimensions confirms the stability of the mapping across random feature samples. Our contributions are three-fold: A general-purpose mapping method that visualizes and formalizes correspondences between embedding features and topic representations; Empirical evidence that text embeddings and topic models are not independent descriptors but can mutually validate each other's semantic structures; A numeric indicator that captures the degree to which embedding features correspond to high-quality topics, providing a new tool for evaluating embedding interpretability and guiding dimensionality reduction choices. These findings suggest that topic-embedding mapping can serve both as a diagnostic for embedding quality and as a means to visualise embedding dimensions more human-interpretable, advancing the practice of collection description in IR.

Keywords: Embedding Interpretability, Language Model Explanability, Topic Modelling

1 Introduction

Text representation is a key task for collection description and organisation in information retrieval (IR) and related areas. Word embedding has become a widely adopted strategy for representing words with fixed-length vectors. However, the representation of longer texts and collections of texts is more challenging than word representations (Incitti et al., 2023).

 WWW: https://irrj.org DOI: 10.54195/irrj.23703

In this work, we study the interpretability of text embeddings for documents by linking them to a fundamentally different but more interpretable collection descriptor, topic modelling.

Topic modelling and text embedding are two approaches to semantic representation of document collections. Although text embedding has been extensively examined as an effective method for capturing semantic characteristics of words and individual documents, topic modelling is more typically used to extract shared semantic information among documents in a collection. The common structure of embedding models usually includes a neural network, an encoding method for the input documents, and a predefined length of output vectors. Some of these models have achieved strong performance in IR tasks, including document classification, query formulation, and recommendation systems (Mikolov et al., 2013; Vulić and Moens, 2015; Ganguly et al., 2015). Due to the effectiveness of embedding techniques in such applications, a variety of refinements have been proposed for conventional topic models using text embeddings (Wu et al., 2020). The applications of text embedding in topic modelling include smoothing the bag-of-words representations as inputs (Seifollahi et al., 2021; Gupta and Patel, 2021); capturing rich semantic features of short texts (Meddeb and Romdhane, 2022; Zuo et al., 2023; Murshed et al., 2022); and constructing new topic model structures with pre-trained language models (Sia et al., 2020; Meng et al., 2022; Jin et al., 2021).

The many approaches for incorporating text embeddings into topic modelling reflect the relatedness of these two techniques. In this paper we argue that the document representations and topical characteristics of the collection loosely correspond to each other and should not be considered as independent sources of information. Instead, we can exploit one to increase the interpretability of the other. If we take a more general view of text embeddings, noting that the primary objective is to learn representations of individual documents, the features also reflect collection-wise semantic characteristics of the collections that were used for training. Likewise, an objective of topic modelling is to capture collection-wise features to describe the documents as a whole. Therefore, studying the connection between the two document representations helps us to understand how the individual dimensions of an embedding capture the semantics of the collection.

A challenge in exploring correlations between topic modelling and text embeddings is that they are assessed under different principles. In IR, a typical evaluation of a model involves a pipeline for assessment and validation. For example, to evaluate the performance of a new text embedding model, it is usually compared with a state-of-the-art baseline model within a series of tasks. However, the use of a pipeline may be an obstacle to understanding the correspondences and differences between distinct kinds of algorithm. There are two reasons why a baseline model is not suitable for this research. First, interpretability of text embedding is not a property that has been defined and quantified explicitly, and a single measure of embedding is insufficient to understand the interpretability of text embeddings in general. Second, a lack of interpretability does not suggest any direct impact on the performance of a model in standard tasks. Therefore, we seek a different approach to study interpretability in a way that can assess text embeddings and topic modellings within a unified narrative of collection description.

To address the challenges in the interpretability of text embeddings discussed above, we design experiments with the following objectives.

- To investigate the relationship between topic modelling and text embedding as document representations for retrieval purposes;
- To improve the interpretability of text embeddings by exploiting the mapping to topics;
- To explore novel approaches to evaluate the effectiveness of collection descriptors beyond a standard model-testing pipeline.

The contributions of this work are three-fold: First, we propose a new mapping method between topic models and text embeddings that visualises the correspondence between topics and embeddings in a range of ways. Second, we demonstrate that topic models and text embeddings are not independent sources of information as document representations, and further that their correspondence can be used to verify the performance of each other. Third, we propose a numeric indicator ΔS, to quantify the degree of correspondence between topic models and text embedding with our proposed mapping method.

The paper is organised as follows: In Section 2, we review relevant literature on topic models, topic specificity, and text embedding. In Section 3, we describe our proposed mapping strategy and discuss the theoretical implications with respect to collection description. The experiments and results are presented in Section 4. Finally, we conclude in Section 5.

2 Related work

We first discuss several state-of-the-art text embedding methods used for collection description and organisation, as well as strategies used for studying embedding interpretability. We also introduce the general principles of topic modelling and how the quality of topics generated by such models is evaluated; and specifically discuss a recent measure, topic specificity, which we introduced in earlier work (Yuan et al., 2023).

2.1 Text embedding

Text embedding aims to transform natural language into high-dimensional vectors. The embeddings have different levels of granularity, with the first embeddings that were proposed focusing on vectorised representation of individual words (Mikolov et al., 2013). More recent embeddings focused on longer text representation (Le and Mikolov, 2014). To differentiate between these two types of embedding, we refer to the former as *word embedding* and those representing longer texts as *text embedding*.

Early embeddings were mostly designed to represent words, where the objective is to represent words as vectors in a high-dimensional semantic space; the number of dimensions of the semantic space can be fixed or user-specific depending on the construction of the system (usually a neural network) used to train the embeddings. These embeddings can be categorised as language models, as they aim to optimise based on the likelihood of generating target sequences of words. As a word embedding is usually dependent on the contextual features around the target word, there is a trend when using large collections to obtain pre-trained embeddings. These can then be further tuned for a specific application. Examples of word embeddings include *Word2Vec* (Kim et al., 2017; Mikolov et al., 2013), GloVe (Pennington et al., 2014), and BERT (Devlin et al., 2019). These embeddings are

pre-trained representations of words that capture semantic features of natural languages (Şenel et al., 2022).

In more recent work, new proposed embedding approaches aim to transform sentences or even passages into vectors of the same length (Le and Mikolov, 2014; Reimers and Gurevych, 2019; Ganesh et al., 2016; Incitti et al., 2023). Le and Mikolov (2014) introduced *Doc2Vec* embedding, which takes the general structure of *Word2Vec* embeddings and adds an additional word-like vector to represent paragraph-wise information. There are also text embeddings that use a pre-trained language model; for example, Reimers and Gurevych (2019) proposed *Sentence-BERT*, which adds a pooling process on top of the BERT structure and yields a fixed-length vector to represent the input sentence. Despite the strong performance of these embeddings in IR tasks such as query expansion, the interpretability remains a challenge in understanding what is captured by the embeddings, in part because text embeddings are constructed from word embeddings that are already difficult to interpret.

Sentence embeddings have been widely applied in IR. *Word2Vec* (Mikolov et al., 2013), which generates semantic representations of words using a skip-gram model, has been used to improve query formulation and refinement (Ganguly et al., 2015). Vulić and Moens (2015) built cross-language retrieval models with their bilingual skip-gram model. The success of text embeddings is because the focus of study in IR is usually on documents; sentence embeddings can convert documents with varying lengths into fixed-length representations. The document embeddings can then be used to calculate distances between documents in multiple downstream tasks, such as clustering (Kim et al., 2017), collection organisation and retrieval (Zuccon et al., 2015; Tanabe et al., 2018; Guo et al., 2022; Zhan et al., 2020), and recommendation (Karvelis et al., 2018; Yang et al., 2016; Hassan et al., 2018).

2.2 Interpretability of embeddings

Interpretibility refers to the users' ability to comprehend decisions made by a machine learning model and to predict the outputs of this model given a certain task (Miller, 2019). Murdoch et al. (2019) defines interpretable machine learning as 'extraction of relevant knowledge from a machine-learning model concerning relationships either contained in data or learned by the model'. In the context of this work, the interpretability of embeddings describes the extent to which users can understand in real-world terms what the dimensions of the embedding capture.

Studying the interpretability of embeddings helps us understand how embeddings capture the semantics of texts and may, for example, assist with model tuning for a specific purpose. To make an embedding interpretable, an intuitive solution is to learn what the individual dimensions represent and what features they capture in terms of the source text. The desirability of interpretability became apparent because it was challenging to explain the performance of the early word embedding models such as Word2vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014), as well as contextualised embeddings such as ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019). These models learn the semantic features of raw texts and output fixed-sized vectors to represent words. The output embeddings serve as base representations for more complex models (Şenel et al., 2022). Studying the interpretability of word embeddings helps us to understand these more complex models.

Techniques for studying the interpretability of embedding can be categorised into two types: task-based interpretation and vector-space interpretation. An example of task-based interpretation is intrusion detection (Prouteau et al., 2022), which involved an evaluation framework for the interpretation of word embedding with a word intrusion task. For the dimension of interpretation of embedding, the set of words with high weights in this dimension must be semantically coherent. Therefore, interpretability is measured by the number of intruder words in the set. More recently this approach has been extended to use word embeddings rather than raw words to achieve task-based interpretation of longer text embeddings. For example, Jha et al. (2023) proposed Contrastive Long Document Encoder (CoLDE), a transformer-based framework to capture document similarity at three different levels of granularity within or across documents. The fine-grained similarity scores generated by the framework not only aid better interpretability of the documents but also support improved understanding of long-text embedding behaviours. CoLDE and other interpretable encoders help with answering one question: When the embeddings of two documents are similar, to what extent can humans understand why they are similar? However, the focus is on interpreting the similarity between documents rather than on interpreting the embedding space.

There is also research that approaches the interpretability challenge by exploring the vector space of the embeddings. Ethayarajh (2019) studied contextualised word embeddings by comparing the geometry of embedding spaces, where the intuition is that for a word in different contexts, the embeddings are different. They used cosine similarity to compute the distance between the two embeddings of a word in different contexts. The significance of this work is that it provides a quantitative measure of semantic difference when the meaning of a word is extended or varied. It served as a foundation to many quantitative studies on embedding interpretability for more complex embeddings driven by large language models (LLMs), and also as an inspiration of this work, on quantifying semantic difference among various embedding approaches at the document-collection level.

Despite the differences in these approaches to evaluation of interpretability, these techniques tend to consider the interpretability of an embedding as binary: that is, if an embedding does not satisfy the criteria of the assessment, it is considered as a non-interpretable embedding method. Nauta et al. (2023) viewed interpretability as a multifaceted characteristic and argued that a quantitative way of measuring interpretability should result in a multidimensional view that indicates the degree to which certain properties are satisfied. However, despite the survey's identification of a promising research direction for visualizing multidimensional embedding interpretability, follow-up work has been scarce, primarily because there is a lack of a suitable semantic proxy for studying embeddings of long texts that is both intuitively human-readable and rigorously quantifiable.

In addition to evaluating interpretability with intrinsic measures, interpretability can also be evaluated by downstream tasks that depend on the interpretability of embeddings. Tasks that require an understanding of text semantics can be used as a proxy to evaluate the performance of text embeddings. For example, Şenel et al. (2022) used sentiment analysis, question classification, and news classification to evaluate their proposed interpretable embedding, bidirectional imparting, or BiImp. However, it is not clear how interpretability contributes to the performance of embeddings in the tasks. Tao et al. (2023) proposed a deep learning-based framework for interpretable text classification, IDeL. The framework

consists of three main components: feature penetration, instance aggregation, and feature perturbation. They also evaluated models constructed by this framework by two tasks, fake news detection and social question categorisation. Methods similar to this one can be considered as a combination of the task-based approach and vector space analysis approach to interpretability.

New embedding approaches have been proposed that are inherently interpretable by controlling the semantic features captured in each dimension. The Word2Sense model uses the LDA topic model (discussed further below) to extract the distributions of word sense and form an alignment to the dimensions of the output embeddings (Panigrahi et al., 2019). Subramanian et al. (2017) proposed sparse interpretable neural embeddings which transforms traditional embeddings such as word2vec into interpretable embeddings based on the denoising k-sparse autoencoder.

While research of this kind is focused on generating interpretable embeddings, there is also work that seeks to use interpretable embeddings for downstream tasks such as text classification. Singh et al. (2023) proposed a novel approach to augmenting interpretable models for text classification with LLMs. They used LLMs for training but not inference so that the models can be trained in a decoupled way and have shown that this method can be used to solve practical tasks such as language zone identification in neuroscience.

A key feature that interpretable embeddings share is vector sparsity. Since the shared goal of these embeddings is to generate interpretable document representations and collection descriptions, higher dimensional vectors can capture more details of the collection than vectors with smaller dimensions. To describe the same amount of information, the interpretable embedding vectors are usually sparser than traditional non-interpretable embeddings. When embeddings are learnt, they are learnt for a certain task. For a language model, the task is to generate word sequences that maximise the likelihood of a target sentence. Therefore, the features learnt need to address the aspect of natural language. The underlying assumption is that the embeddings remove and reorganise the original features in the natural-language representation and generate new features that better suit the task. In contrast, non-interpretable embeddings are usually pre-trained for general purpose use. They should reflect the characteristics of the source collection without obvious bias towards any aspect of the semantics.

Overall, while these interpretability studies have considered significant aspects of how embeddings encode information, a unified, scalable framework for long texts is still absent.

Our approach addresses this gap by combining vector-space and task-based perspectives into a visual framework of embedding interpretability and can serve as a basis for more transparent and tunable embedding methods. Unlike studies that focus on improving or benchmarking embedding models, our framework is embedding-agnostic: it operates on any document-level vector representation and evaluates how such embeddings can be interpreted through their correspondences with topic models, rather than on their downstream performance.

2.3 Topic modelling

Topic modelling is an approach to learning representative themes from collections of documents. Early topic modelling approaches include statistical models such as latent semantic

indexing (LSI) (Deerwester et al., 1990), non-Negative Matrix Factorisation (NMF) (Gillis and Vavasis, 2014; Wang and Zhang, 2012), and singular value decomposition (SVD) (Zheng and Wang, 2022; Steinberger and Ježek, 2005). These models follow a shared principle: the difference between term frequency within a document and term frequency in the collection as a whole is used as an indication of topicality. The limitation of these models is that topic modelling based on term frequency is coarse, so the performance of such models can be poor when topics are used in downstream tasks in which specific semantics is required.

Latent Dirichlet Allocation (*LDA*) (Blei et al., 2003), which is a representative and widely used probabilistic topic model, addresses the limitation of statistical models. In contrast to statistical models, *LDA* does not rely solely on the decomposition of the document term matrix; instead, it assumes that the distribution of terms is conditioned on a set of latent topics and that the distribution of topics is conditioned on the collection of documents (Blei et al., 2003). For many years *LDA* has been the most robust topic modelling approach as evaluated by topic coherence (Newman et al., 2010; Mimno et al., 2011; Morstatter and Liu, 2017), topic diversity (Bu et al., 2021), and perplexity (Blei et al., 2003).

Recent research seeks to use word embeddings and other forms of neural networks to improve traditional topic modelling methods, including *LDA*, by providing richer forms of input documents and topical structures. Cheng et al. proposed bi-term topic modelling (BTM), which learns term co-occurrence generation patterns instead of single-term sequence patterns in order to resolve the sparsity issue of term co-occurrence in short texts. BTM replaces the original term representations with embeddings that consist of multiple words and has achieved better topic coherence and diversity compared to several baseline models.

There has been a trend of using pre-trained language models on top of the conventional topic models to solve some particular problems: Han et al. proposed the unified neural topic model by utilising contrastive learning with conventional term-based representations and pre-trained language models to detect keywords from semantically coherent clusters; Gaussian *LDA* (Das et al., 2015) replaced the original term representation in conventional *LDA* with word embeddings drawn from a multivariate Gaussian distribution; and in the CluWords topic model (Viegas et al., 2019), the term representation in Non-negative Matrix Factorisation (NMF) is replaced by the representation of the nearest term from a pre-trained word embedding model to form a meta-representation of the documents, which enhances the representation of both syntactic and semantic information. The state of the art of embedding-enhanced topic models, *BERTopic*, generates topics by clustering the document embedding representations learnt from a pre-trained language model and calculating class-based TF-IDF vectors as the topic representation for each cluster (Grootendorst, 2022).

2.4 Topic specificity

Our recent work on topic modelling evaluation provides a theoretical foundation for assessment of topic quality from the perspective of document representation. To evaluate the degree to which a topic is representative of a subset of documents in a collection, Yuan et al. (2023) proposed the *topic specificity* measure. Topic specificity can be used on any topic model that represents documents as a vector of percentages or weightings of topics, including conventional topic models such as *LDA* and neural topic models such as *BERTopic*.

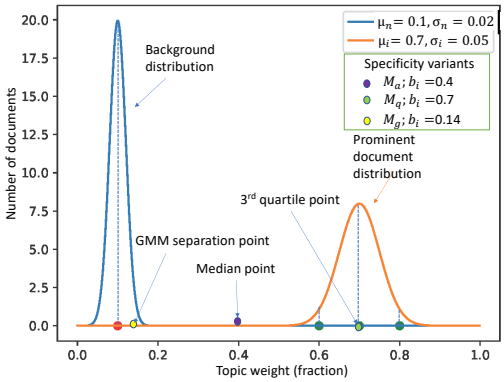


Figure 1: An illustration of the *bimodality* of topic weight distribution over a collection. This figure is reproduced from Yuan et al. (2023).

The derivation of this measure is associated with the phenomenon of *bimodality* that generally exists in topic models. Figure 1 shows how bimodality appears when an ideal topic is generated from a document collection: when a topic is used to describe a collection, the topic only shows high weights in those truely relevant documents while existing as background noise in the rest of the collections. By plotting topic weight of individual documents against the number of documents, a topic with good quality usually shows two peaks in the distribution plot - we call this shape *bimodality*.

Topic specificity is a quantification of the *bimodality* phenomenon. The intuition behind topic specificity is that a topic that is descriptive should have high weights in a small subset of the documents and low weights for the remainder of the documents. If we consider the topic weights of a single topic over the collection as a distribution, we should be able to find a threshold (the base) that differentiates the described subset from the background collection. Adopting the notation of the original paper, the specificity S of an arbitrary topic t_i is defined as

$$S(t_i) = \frac{\mu_i}{1 - b_i} \tag{1}$$

where μ_i is the average distance between prominent documents and the base and b_i is the base of the weight distribution of the topic. High specificity means that the topic clearly discriminates between the subset in which it is prominent and the remainder of the collection.

The base b_i can be calculated using the median of the topic's weight distribution over the collection, the third quartile of the distribution, or a Gaussian Mixture Model (GMM). The way we choose is by GMM since this is a complete mathematical simulation of distribution mixtures without any manual intervention. Hence, for a background distribution $X_n \sim \mathcal{N}(\mu_n, \sigma_n^2)$, the base b_i and the mean μ_i are defined as

$$b_i = 2\sigma_n + \mu_n \tag{2}$$

$$\mu_i = \frac{\sum_{d \in D_i} (f_i^d - b_i)}{|D_i|} \tag{3}$$

where D_i is the prominent document set in a document collection D for topic t_i and f_i^d is the weight of topic t_i for each document $d \in D_i$.

The range of topic specificity scores is 0–1, where 0 means that the topic is not descriptive of any document, and 1 means that the topic is only representative of a specific document. Values between 0.2 and 0.4 are generally observed to be high topic-specificity scores.

Specificity is more sensitive to noise in the topics than other coherence-based measures, which makes it suitable for use in contexts where robust sensitivity to topic quality is required. In addition, topic specificity quantifies how individual documents are described by a limited number of topics, directly capturing the 'aboutness' of topics on collection level. These properties make topic specificity a valid measure of topic quality for our use case and a reliable proxy for embedding semantic interpretation.

3 Methodology

We describe our approach in this section. First, we describe the document collections and models used for the experiments. Then, we use an illustration of embedding and topic model representations of document collection to explain the logic behind the correspondence study. Next, we describe our proposed mapping approach, a quantitative measure of mapping quality, and a feature sampling approach for the stability analysis of the mapping performance.

3.1 Collections

In our experiments, we used three document collections. The smallest document collection contains 5,000 documents and the largest 20,000. The following describes these collections and their corresponding topics:

WIKI**:** A sample of 5,000 documents from a Wikipedia dump. We randomly selected 1,000 documents from five categories each to create our labelled dataset. The categories chosen are culture, geography, health, history, and mathematics.[1]

20NG**:** Contains 18,846 documents that were collected from the BBC News website. The documents are grouped into 20 news categories (Lang, 1995). [2]

WSJ**:** Contains 98,733 Wall Street Journal articles collected as part of the TREC disks (Voorhees and Harman, 2005). Due to computational limitations, we randomly selected 20,000 documents from the collection. This is an unlabelled collection, and hence the natural number of topics is unknown. [3]

1. Categories are selected as in `https://en.wikipedia.org/wiki/Wikipedia:Contents/Categories`. Pages were crawled on 2 September 2022. The dataset is available upon request.
2. 20 News Group dataset is available at `http://people.csail.mit.edu/jrennie/20Newsgroups/`
3. The Wall Street Journal (WSJ) collection used in TREC evaluations is part of the proprietary Tipster and TREC disks distributed by the Linguistic Data Consortium (LDC). It is not publicly available and requires a license agreement for access.

3.2 Model selection

For text embeddings, we select the models that are particularly designed for sentence or long-text vectorisation, namely *Doc2Vec* and *Sentence-BERT*. Note that there are many embedding approaches that have outstanding performance in IR tasks, but they are mostly term-based embeddings and not suitable for the task at hand. For fair comparison and a deeper illustration of the alignment between topics and embedding features, we generate three types of sentence-based embeddings in our experiments.

***Doc2Vec* Le and Mikolov (2014).** An extension of the *Word2Vec* model, which was described earlier in Section 2.1. In addition to word-level modelling, *Doc2Vec* adds another vector that captures the sequence characteristics of the sentence in the document as part of the input. To train *Doc2Vec* embeddings, the documents need to be tokenised and lemmatised first and the number of dimensions needs to be pre-defined. The default number of dimensions is 400.

***Sentence-BERT*.** An extension of the conventional BERT model, which was described in Section 2.1. It uses a pre-trained language model as the input to generate document-level embedding vectors. Unlike *Doc2Vec*, the dimensionality of *Sentence-BERT* embeddings is fixed and depends on the pre-trained language model used for training. The number of default dimensions is 384.

***RepLLaMa*.** A retrieval-enhanced embedding method built on the LLaMA transformer architecture (Ma et al., 2024). During fine-tuning, RepLLMa is trained with a document-level retrieval objective so that its hidden-state representations capture rich semantic alignments across documents. The model outputs fixed-length vectors (e.g., 4096 dimensions). For fair comparison, the dimensions of embeddings are reduced to a range between 10 and 200 using random projection. This adjustment prevents performance differences from being confounded by representational capacity and aligns the embedding dimensionality with the scale of the topic space. That is, for the selection of collections, the number of topics is much lower than the dimension of *RepLLaMa*, 4096. In addition, using random projection preserves pairwise distances between documents. According to the Johnson-Lindenstrauss lemma (Johnson and Lindenstrauss, 1984; Bingham and Mannila, 2001; Achlioptas, 2003), dimensionality reduction for extremely sparse high dimensional space may be achieved without substantial loss of geometric information.

3.3 Text embedding features

As introduced in Section 2.1, text embeddings are document representations that interpret documents of varying lengths as vectors of fixed dimensions. Intuitively, for a set of text embeddings learnt from the same collection, we can interpret the dimensions of the embeddings as the signature facet shared by all documents from a document collection. If we learn a set of embeddings for a document collection, we can represent the collection as a set of feature vectors. Figure 2 shows an illustration of a document-feature matrix that represents the entire collection. The rows are text embeddings of the documents; the columns represent the semantic aspects shared by all the representations of the documents. In other words, the column vectors are the semantic characteristics of the collection described by the document embeddings. From now on, we refer to these collection representations learnt

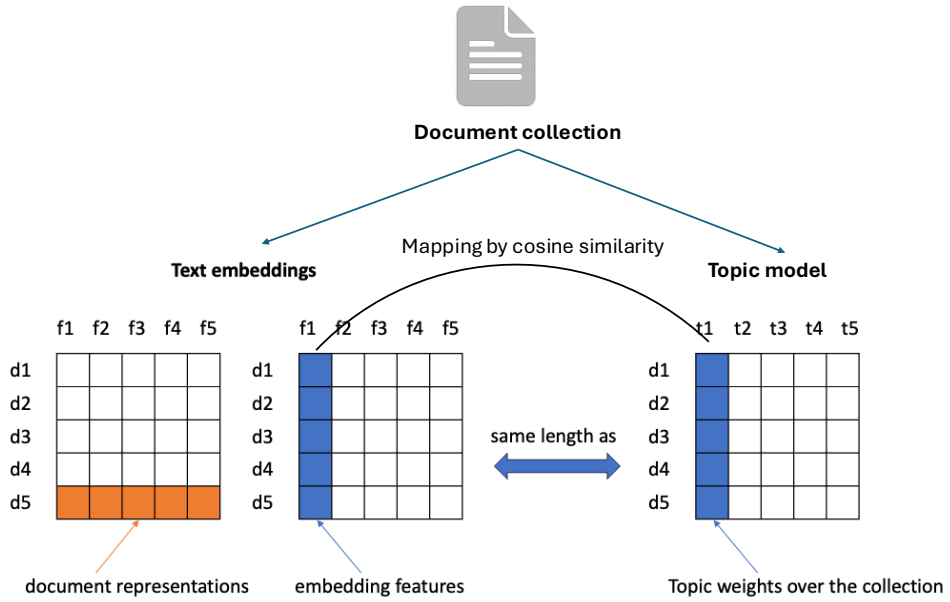


Figure 2: Both the text-embedding matrix and the topic model are derived from the same document collection. Unlike conventional use of text embeddings where document vectors (rows) are compared, this mapping operates on the embedding-feature vectors (columns) of the embedding matrix. Each embedding feature forms a vector over the document collection, having the same dimensionality (number of documents) as a topic vector from the topic model. To align the two representations, we compute the cosine similarity between each embedding feature and each topic vector, identifying the closest pairs as feature–topic matches that characterize semantic correspondences between the embedding and topic spaces.

by embeddings as *embedding features*. In previous attempts to create interpretable embeddings, the dimensions of the embedding vector are usually encoded as collection-wise topics or concepts. This is a reasonable assumption based on the expected utility of the embeddings. If the assumption is correct, we should be able to observe these topics from the collection regardless of the embedding method, that is, for the embeddings that learn the features implicitly from the collection, the dimensions should also reflect a semantic aspect of the collection.

Assuming that these embedding features are representations of collection topics implies that the embedding features should have a large overlap with other forms of topic representation, such as LDA topics. However, LDA topics and embedding features are produced by completely different principles and mechanisms: the document-topic representations are derived from probabilistic inference based on the topic-term distributions of topic models while the embedding features are learnt by the likelihood of a target sequence of words described in a fixed number of dimensions.

A key factor that differentiates these two types of collection representation is interpretability; topics can be interpreted as the distribution of topic weights over the collection, whereas the embedding features are simply real-valued numbers in a high-dimensional space.

This can be verified by a bimodality check: If we take the weights of a topic in a document collection and plot them as a distribution, a good topic should have a bi-modal shape. However, for dense embeddings, if we take the values of an embedding feature in the same collection and plot the distribution, the embedding features usually have a normal distribution. The bimodal distribution is what the sparse embeddings are intended to achieve. In other words, a single feature should describe a small subset of the documents, while being irrelevant to most of the documents in the collection. These features are also intended to be disjoint, so that they only matter to certain documents if the documents contain the semantic information described by the features.

Hence we conclude that topics and embeddings are not equivalent – an unsurprising conclusion perhaps, but also one that provides a basis for the next steps in our investigation.

3.4 Topic-embedding mapping approach

We propose mapping methods between embedding features and topic models. For a document collection $\mathcal{D}$, we learn an embedding representation $\mathcal{V}$ with shape $|\mathcal{D}| \times |\mathcal{V}|$. Then we train a topic model T and infer the document-topic representation with shape $|\mathcal{D}| \times |\mathcal{T}|$. As explained in the previous section, there are $|\mathcal{V}|$ embedding features and $|\mathcal{T}|$ topical representations learnt to describe the same collection.

Mathematically, both $\mathcal{V}$ and $\mathcal{T}$ can be regarded as projections of the same set of documents $\mathcal{D}$ into different lower-dimensional subspaces. Each subspace encodes the same underlying semantic structure, though derived through different optimization objectives. To quantify the similarity between these subspaces, the comparison metric should depend on the directional alignment of their vectors rather than their absolute magnitudes, since both embeddings and topic weights can be arbitrarily scaled. Therefore, cosine similarity is used, as it measures the angular proximity between vectors, capturing semantic alignment while being invariant to scale. This makes it particularly suitable for comparing representations that express relative importance or association strength, as in the case of embedding features and topic distributions.

The *mapping* between topics and embedding features is the process of finding the dimensions of the embedding space and the topical space that are close to each other. For a set of candidate embedding features and a set of candidate topics, we define a feature-topic pair as a mapping if the candidate embedding feature and the topic have the largest cosine similarity of any pair formed in the candidate sets. According to the mapping, we can categorise the topics of a topic model into two types: *mapped topics*, which occur in at least one pair of the mapping; and *un-mapped topics*, which are topics that are not in any pair of the mapping.

The mapping is conducted on the basis of embedding features. For each feature from the embedding representation $v \in \mathcal{V}$, first compute the cosine similarity with each topic from the topic model $t \in \mathcal{T}$; then map feature v with topic t if it has the largest cosine similarity with feature v, that is, $\text{argmax}_{t \in \mathcal{T}}\, cosine(v, t)$; and add feature v to the mapping set of topic t and remove v from the embedding representation, that is, $\mathcal{V} \leftarrow \mathcal{V} - v$. This mapping is performed without applying a similarity threshold, as the goal is to assign every embedding feature to the most related topic, thereby ensuring complete feature coverage for interpretability analysis rather than filtering out weak associations. Theoretically, one

topic can be mapped to zero or more embedding features. The one-to-many nature of the mapping contributes to the interpretability of embedding features.

Feature Vector Sparsity Justification. One of the common observations w.r.t. embedding features is that they are highly sparse. Intuitively, it seems inappropriate for a very sparse vector to be mapped to a dense topic vector. However, feature vectors and topic vectors are representations of different spaces; the sparsity of feature indicates the effectiveness of this feature in representing a particular aspect of a small subset of the collection. The features are supposed to be sparse for the optimal descriptiveness of the collection. An equivalent quantifier of the sparsity of embedding features is the bimodal shape of the topics of a topic model. An informative topic should be able to differentiate the documents that belong to it from the rest of the collection. If we consider the topic weight distribution over all documents in a collection, it is highly skewed; for most of the documents, the topic is irrelevant and therefore has a weight very close to zero; for a small subset of documents, the topic is the dominant topic and hence has very large weight. Now, if we take the topic weights as a vector, it is also a sparse vector. But the representation of the topic, the distribution of word weight in a topic, is always a dense vector, regardless of the specificity of the topic.

3.5 Mean mapping specificity difference

In the previous section, we explained how the mapping mechanism relates the embedding features to topics. If there is a good mapping, this can aid in the interpretability of text embedding. To demonstrate how this mapping can aid interpretability, we use a variant of topic specificity, which measures the extent to which a topic represents a particular characteristic of the document collection.

To measure the quality of a topic model with specificity, we calculate the mean specificity scores over a set of topics generated by a topic model. Intuitively, if the features learnt from the text embedding approach are more representative than the conventional topical features, the mapped topics of these features should have higher specificity than the overall specificity of all topics. Hence, we can quantify the increase in topic specificity by computing the difference between the mean specificity of topic sets that were originally generated by the topic model and the topic sets after mapping using the embedding features.

Using a topic model $\mathcal{T}$ as a reference, the quality of a set of embedding features can be expressed by the mean mapping specificity $\mathcal{S}'$ as follows:

$$\mathcal{S}' = \frac{\sum_{t \in \mathcal{T}'} \mathcal{S}_t \times \mathcal{N}_t}{\sum_{t \in \mathcal{T}'} \mathcal{N}_t} \tag{4}$$

where $\mathcal{N}_t$ is the number of features of the embedding representation mapped to topic t, $\mathcal{T}' \subseteq \mathcal{T}$ is the set of topics that have at least one mapped embedding feature, and $\mathcal{S}_t$ denotes the specificity score for topic t.

We can determine how using only the mapped topics can impact the specificity of the topic model. At one extreme, all features can be assigned to the topic with the highest specificity; conversely, at the other extreme, all features can be mapped to the topic with the lowest specificity. These two extremes create a boundary around how the mean specificity of the topic model can vary after the mapping is performed. Therefore, we can derive a

measure of the alignment between text embeddings and topics within a bounded range of $[-1, 1]$. We call this measure the *Mean Mapping Specificity Improvement Rate*, denoted by ΔS, and define it as follows:

$$\Delta\mathcal{S}(\mathcal{T}', \mathcal{T}) = \begin{cases} \dfrac{\mathcal{S}' - \mathcal{S}}{\max_{t\in\mathcal{T}}(\mathcal{S}_t) - \min_{t\in\mathcal{T}}(\mathcal{S}_t)}, & \text{if } \max_{t\in\mathcal{T}}(\mathcal{S}_t) \neq \min_{t\in\mathcal{T}}(\mathcal{S}_t); \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

where $\mathcal{S}$ and $\mathcal{S}'$ denote the mean specificity over the topics in $\mathcal{T}$ and $\mathcal{T}'$, respectively. ΔS quantifies the degree of alignment between embedding features and topic specificity: a positive value indicates that more features are mapped to topics with above-average specificity, while a negative value suggests that features are more likely mapped to topics with below-average specificity. When all topics have identical specificity (i.e., zero variance), the denominator becomes zero; in such cases, the measure is set to 0, as no improvement in specificity can be defined.

3.6 Embedding feature sampling

To assess the stability of ΔS under different random projections, we vary the number of projected dimensions and repeat the projection with different random seeds. This measures how robust the mapping from embedding features to topics is when the embedding space is randomly re-parameterized.

We use the same three corpora and embedding methods (*Sentence-BERT*, *Doc2Vec*, *RepLLaMa*) as before. Each original embedding matrix is $E \in \mathbb{R}^{D\times N}$, where D is the number of documents and N the full embedding dimension. Let $\mathcal{F} = \{f_{\min}, f_{\min} + s, \ldots, f_{\max}\}$ denote the range of target dimensions (e.g. $f_{\min} = 10$, $f_{\max} = N$, step $s = 10$).

For each $f \in \mathcal{F}$, we perform B independent random projections:

1. Generate a random projection of E into f dimensions using a different random seed for each draw, producing $E_f^{(i)} = \text{RandomProj}(E; f, \text{seed} = i)$.

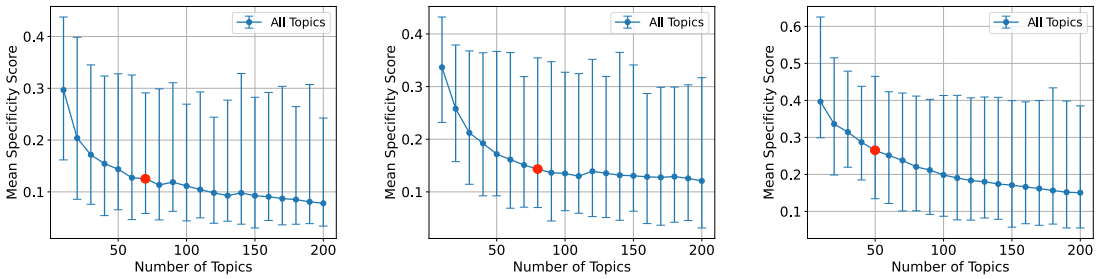
2. Apply the mapping procedure from Section 3.4 to align $E_f^{(i)}$ with the chosen topic model, yielding a set of (feature, topic) mappings.

3. Compute ΔS on the mapped topics and record the score for this projection.

This yields B ΔS values for each f. We summarize the results by reporting the mean and standard deviation of ΔS over the B random seeds, thereby quantifying mapping stability as a function of projection dimension.

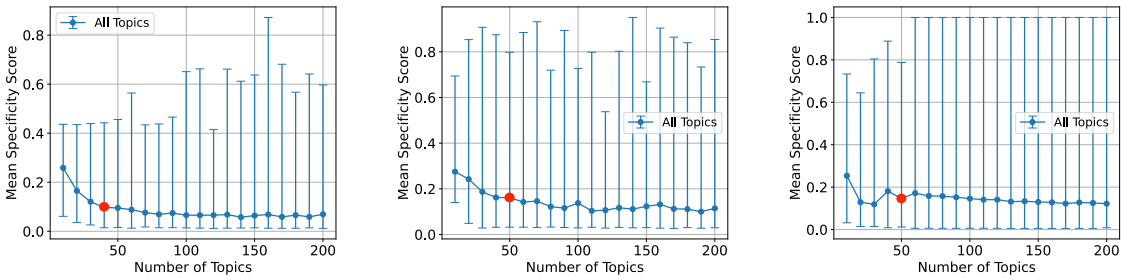
4 Experiments

With the proposed mapping method and the feature sampling approach described in Section 3, we design a set of experiments to reveal how topic models contributes to the interpretability of text embeddings. As for all the visualisation tasks of semantics of texts, the experiment processes include both quantitative measurements and visual interpretations. It ensures the interpretability study to generate not only numbers but also human-interpretable outcomes.



(a) Optimal $k = 70$ for 20NG. (b) Optimal $k = 80$ for WIKI. (c) Optimal $k = 50$ forWSJ.

Figure 3: Average topic specificity over all topics when ranging over number of topics for *LDA*. The curve's turning point marks where the mean specificity score levels off, indicating that adding more topics beyond this point yields diminishing returns and primarily introduces noise, as evidenced by subsequent drops in specificity. While the choice is somewhat arbitrary within this acceptable range, we found it sufficient for our purposes. We elaborate on this decision in the section *Experiments*.



(a) Optimal $k = 40$ for 20NG. (b) Optimal $k = 80$ for WIKI. (c) Optimal $k = 50$ for WSJ.

Figure 4: Average topic specificity over all topics when ranging over number of topics for *BERTopic*. The optimal number of topics is determined in the same way as for *LDA*. (In Plot (c) on WSJ, the top topic always scores 1—reflecting a single very specific core topic—but this does not affect our elbow choice, since we rely on the mean curve rather than the max/min scores.)

The experiments can be roughly divided into three components: Determination of the optimal topic model; Visualisation of topic evolution and mapping; Stability analysis of embedding semantics. These components are described in sections 4.1, 4.2, and 4.3.

4.1 Determining optimal numbers of topics

To show that our proposed mapping approach is valid for most topic models, the two we select are the most representative in their categories: *LDA* is the most commonly used statistical topic model and the foundation of many derivatives; *BERTopic* is a representative of the state-of-the-art neural topic models using pre-trained embeddings as input.

Due to the different nature of these topic models, they usually converge to different numbers of topics, k. Hence, we use a slope-plot with respect to topic specificity to determine the optimal number of topics for each selected topic model. First, we do a grid-search over the topic models by varying the number of topics from 10 to 200 with a step of 10 for both models in all collections. Next, for each run of the model, we calculate the specificity score per topic and plot the mean specificity per topic model run against the number of topics. As the number of topics increases, the topic model will have more noise topics with low specificity scores; even the topic with high quality will break, and eventually all topics end up with low specificity scores. Therefore, we end up with a decline of specificity scores. The last thing is to determine the optimal number of topics using the slope. The tuning point of the curve represents the optimal number of topics for the chosen topic model and collection. It reflects a balance where the number of coherent topics is sufficiently objective without being excessive, while noisy topics begin to separate out from the bulk of meaningful topics.

The precise location of the turning point is not critical, especially when the slope of the curve is relatively flat. In such cases, there often appears to be a range of values that can reasonably be considered part of the turning region. Selecting any representative value within this range is sufficient, as the exact number of noisy topics has limited impact on the subsequent mapping. In contrast, the number and quality of coherent topics play a more important role in determining the effectiveness and interpretability of the embedding-to-topic alignment.

The results of *LDA* and *BERTopic* of each collection are shown in Figures 3 and 4. For *LDA*, the optimal numbers of topics for collection 20NG, WIKI, and WSJ are chosen at 70, 80, and 50, respectively. For *BERTopic* model, the optimal numbers of topics are 40, 80, and 50. We will use these model runs for the following mapping experiments.

4.2 Embedding interpretability by mapping

We now examine the correspondence between the topics and the embedding features. An intuitive assumption of embedding features is their high semantic granularity compared to topics. Therefore, for a topic of good quality, the amount of information represented by the topic can be decomposed into multiple embedding features. We argue that mapping can help us identify topics that are more representative of individual documents, since topics with mapped features are closer to text embeddings than are other topics. Our aim is to demonstrate that, regardless of the number of features, which can be smaller or larger than the number of topics, features tend to map to topics with higher specificity, and at times multiple features are mapped to a topic that best captures document-level characteristics.

Intuitively, when there is an equal number of features and topics involved in the mapping, some topics will be left out due to their lack of similarity to any of the features. However, when there are fewer features than topics, it is possible that each feature is mapped to a different topic. In that case, the mapping would fail to distinguish the 'good' topics from the rest. This phenomenon is consistent for various number of topics and embedding features on all three collections. Note that topics with high specificity scores that are not mapped to any features are close to an already mapped topic, and therefore the mapping approach could not allocate a feature to them.

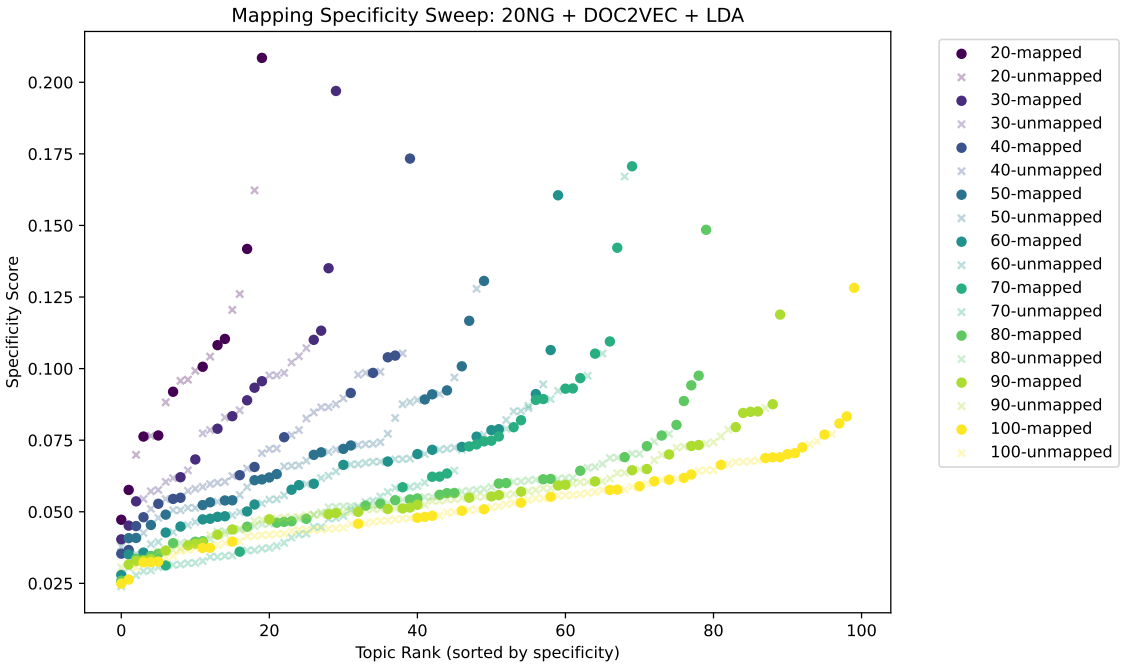


Figure 5: An example of using the mapping to visualise topic evolution. A series of *LDA* models of 20NG collection are mapped to the *Doc2Vec* embedding features of the same collection. When the number of topics is small, topic specificities are very high, but the mapping indicates the topics are likely to be too broad compared to the semantic dimensions captured by the embeddings. As the number of topics increases, topic specificities are dropping gradually due to the decomposition of large topics, yet the features tend to map to the 'good' topics towards the right upper corner of the plot.

Figure 5 illustrates use of the mapping framework to visualize topic evolution, in which we apply *LDA* models to the 20NG collection and map them to the *Doc2Vec* embedding features derived from the same corpus. When the number of topics is small, the resulting topics exhibit high specificity scores. However, the mapping reveals that these topics are often overly broad relative to the semantic dimensions captured by the embeddings. As the number of topics increases, specificity scores gradually decline due to the decomposition of large, general topics into more fine-grained ones. Nevertheless, the mapped features tend to concentrate in the upper-right region of the plot, indicating that they consistently align with a subset of coherent topics even as the overall topic structure becomes more granular.

We present mapping results for both topic models in Table 1. Based on the optimal number of topics selected by the elbow method in the previous step, we random project each embedding and reduce them to the same number of features for each collection. The mapping is then conducted between the topic models and embedding features for each collection. To illustrate the trend, Figure 6 provides an example of the mapping between *LDA* and *Sentence-BERT* and the mapping between *BERTopic* and *RepLLaMa*.

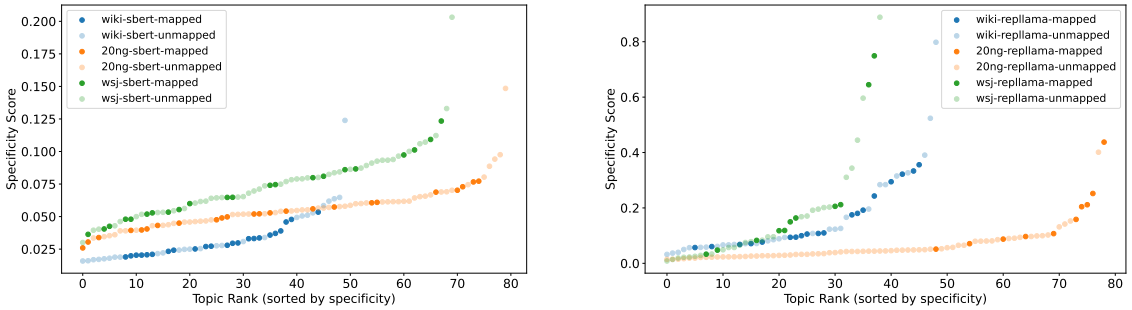
These results demonstrate the interpretability of embeddings in two ways. First, mapped topics usually have higher specificity scores, suggesting better descriptiveness of document

Collection	Topic model	Embedding	Mapped	Unmapped	Δ
WIKI	*LDA*	*Doc2Vec*	0.105	0.064	0.040
		Sentence-BERT	0.111	0.062	0.049
		RepLLaMa	0.118	0.066	**0.053**
	BERTopic	*Doc2Vec*	0.140	0.129	0.011
		Sentence-BERT	**0.158**	0.109	0.049
		RepLLaMa	0.134	0.118	0.016
20NG	*LDA*	*Doc2Vec*	0.097	0.051	0.046
		Sentence-BERT	0.095	0.045	0.051
		RepLLaMa	0.118	0.052	0.066
	BERTopic	*Doc2Vec*	0.200	0.059	0.141
		Sentence-BERT	0.191	0.058	0.132
		RepLLaMa	**0.208**	0.070	**0.138**
WSJ	*LDA*	*Doc2Vec*	0.105	0.078	0.027
		Sentence-BERT	0.129	0.075	0.054
		RepLLaMa	0.133	0.080	0.053
	BERTopic	*Doc2Vec*	0.183	0.143	0.041
		Sentence-BERT	0.264	0.153	0.111
		RepLLaMa	**0.341**	0.135	**0.206**

Table 1: Average topic–specificity scores (mapped vs. unmapped). Differences of average specificity of mapped and unmapped topics are shown in column Δ.

collection. Since the embedding features are mapped to these topics, the mapping results can be considered as validation of the embedding descriptiveness. Second, mapping multiple embedding features to the same topic suggests greater coverage of the semantics of a topic than an embedding feature. As embeddings generally have many more dimensions than the number of topics in a topic model, the granularity of information is higher in the embeddings. Consequently, to describe a concept in a document collection, text embeddings can use multiple features to decompose the concept. Therefore, a potential way to utilise the mapping results is to summarise what the embedding features describe by visualising the topics to which they are mapped. The topics can also be used to group the embedding dimensions into categories and serve as labels for better interpretability.

As a qualitative case study, we provide examples of mapped and unmapped topics in Appendix A to illustrate how the proposed mapping framework manifests in practice. For instance, in the WSJ collection, topics such as *"president, executive, chief, vice, officer, chairman, name, director, board"* and *"share, stock, trading, dividend"* are frequently mapped to Sentence-BERT features, indicating coherent, collection-relevant semantics that align with the quantitative ΔS results. In contrast, unmapped topics such as *"food, restaurant, franchise"* or *"insurance, policy, premium"* exhibit weaker internal coherence and



(a) Mapping between *LDA* and *Sentence-BERT* embedding features.

(b) Mapping between *BERTopic* and *RepLLaMa* embedding feartures.

Figure 6: Example visualisations of topic-model-to-embedding mappings. The number of scatter points for each collection corresponds to its optimal number of topics.

lower specificity scores. These examples complement the quantitative analysis by providing intuitive evidence that mapped topics tend to be more interpretable and semantically consistent, supporting our claim that the mapping can serve as a diagnostic tool for embedding interpretability.

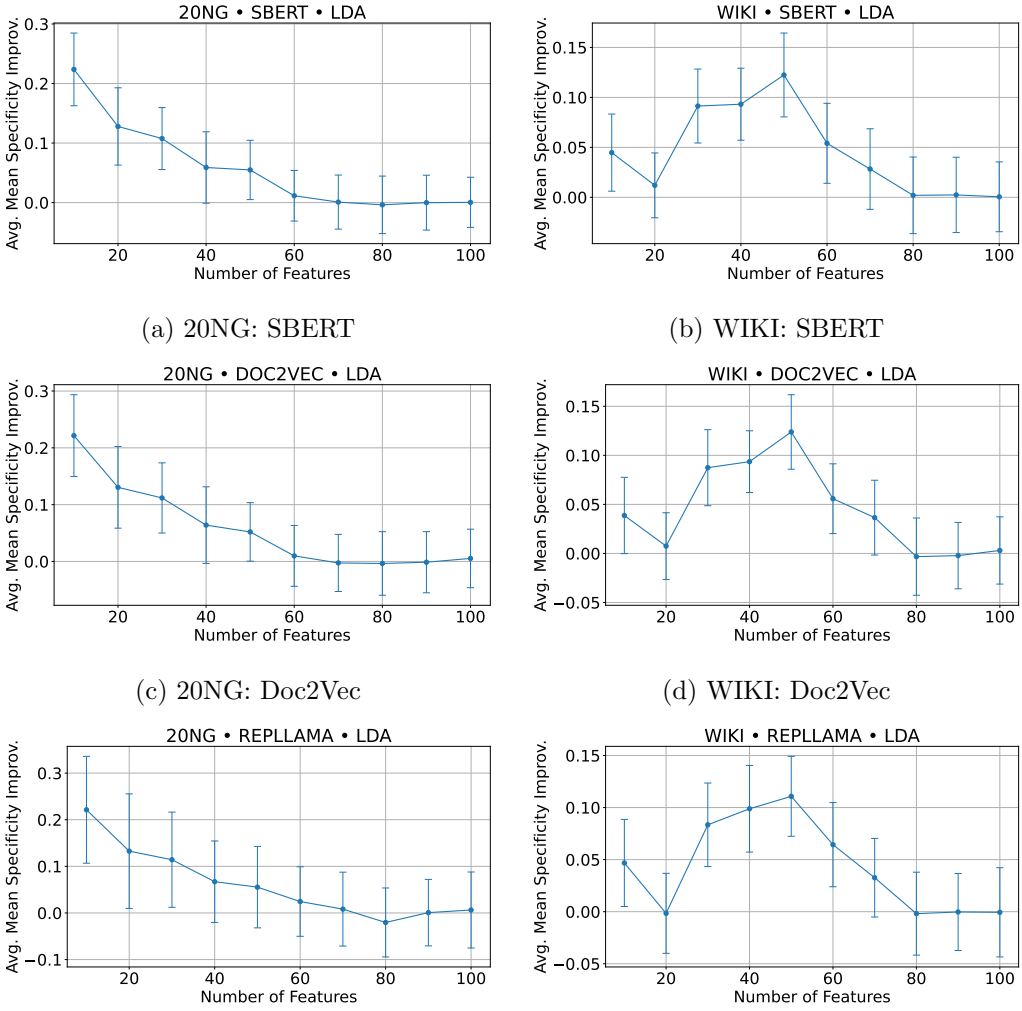
4.3 Stability analysis via feature sampling

The consistency of the mapping outcome is essential to the interpretability of embedding features. With the goal of generating human-interpretable representations of embedding features, the mapping to a topic model should generate robust and reproducible results so that the mapped topics can describe semantics of the embedding features being represented. To address this concern, and to conclude our exploration on the correspondence between topic models and text embedding, we seek to show the stability of the mapping approach by varying the number of sampled features via the method described in Section 3.6.

After sampling features with random projection for 100 runs on each settings of the three embeddings, we found that the topic model might be the ceiling of the interpretability of embedding features. Surprisingly, the optimal number of features is highly dependent on the topic model and less relevant to the embedding method – that is, there is likely to exist a limit on the granularity of semantic interpretability that a topic model can offer for a given document collection, regardless of the embedding methods used.

As the pattern seems consistent among these collection and topic model combinations, we present two of them as an example. Figure 7 show the mean and a min-max range of ΔS over 100 samples of embedding features with varying feature dimensions from 10 to 100 on 20NG collection and WIKI collection. Each subplot represents the trend of ΔS of a embedding method. It is clear that the trend is nearly identical among embedding methods while differing between collections. For 20NG, the peak of mapping performance occurs when embedding features are reduced to 10 dimensions, compared to 50 dimensions for WIKI. Note that the topic models used for this mapping experiment are consistent throughout all variations on the number of features.

Our stability analysis reveals that each topic model enforces its own semantic grain size on a collection, meaning that the optimal embedding dimensionality reflects the model's capacity to distinguish specific topics without introducing noise. As for future study, we will explore whether structured dimensionality-reduction methods, such as PCA or encoder-based models, can sharpen this stability peak and further improve interpretability. We also plan to apply our stability framework to alternative topic modeling paradigms (for example, non-negative matrix factorization or hierarchical Dirichlet processes) to uncover algorithm-specific limits on semantic granularity and guide model selection.



(a) 20NG: SBERT

(b) WIKI: SBERT

(c) 20NG: Doc2Vec

(d) WIKI: Doc2Vec

(e) 20NG: RepLLaMA

(f) WIKI: RepLLaMA

Figure 7: Trend of ΔS when mapping to the optimal *LDA* model with varying feature dimensions on (left column) 20NG and (right column) WIKI collections over 100 samples by random projection. Rows correspond to embedding models: (1) SBERT, (2) Doc2Vec, and (3) RepLLaMA.

5 Conclusion

Text embeddings have long been studied as a tool for collection description and organisation, but embedding interpretability remains a challenge, particularly at the collection level. There isn't a reliable correspondence measure between text embeddings and human-readable representations. However, recent advances in topic modelling evaluation have provided us with a tool to measure the descriptiveness of topics. To our knowledge, no prior work has considered comparing topic models and text embeddings as distinct representations of document collections or explored how they correspond.

In this paper, we propose exactly such a proxy: a robust, collection-level semantic representation derived from topic-model mappings, coupled with a specificity metric that is both human-interpretable, via topic keywords, and quantitatively tractable. The mapping allows us to conduct a systematic comparison between topical representation and embedding representations of the same collection. The mapping confirms as correspondence between topic modelling and text embedding as collection description tools. We have proposed a numerical indicator of mapping performance, *mean mapping specificity improvement rate*, which captures the change in the mean specificity score over mapped topics to that of the original set of topics. This indicator, with values ranging from $[-1, 1]$, can help us to identify embedding features that correspond to descriptive topics and hence enhance the interpretability of embedding approaches.

This work has three contributions. First, we proposed a mapping method between topic models and text embeddings that contributes to the interpretability of embedding features with an arbitrary number of dimensions. Second, we show that topics and text embedding are not two independent sources of information when used as document representations by using *embedding-based mapping* between topics and embedding features; topics can be used as a proxy to infer conceptual information captured by embedding features. Third, we successfully quantify information loss in embeddings using the dimension reduction technique with the mean mapping specificity improvement rate; the measure can be used for comparison between different embedding techniques for semantic studies. These should allow selection of topics for tasks such as collection or document annotation, and can plausibly provide verification that the generated embeddings and topics are of good quality.

However, this approach also has limitations. The quality and interpretability of the mapping are dependent on the topic model's coherence. Poorly defined topics can lead to misleading correspondences. Moreover, biases inherent in the dataset may propagate through both topic modelling and embedding representations, influencing the observed relationships and specificity scores. Future work should explore techniques to mitigate these effects, such as bias correction and topic refinement strategies.

Overall, by mapping embeddings to topic models, we create a multidimensional interpretability visualisation framework that is a step beyond simple binary judgements on interpretability, scales to longer texts at collection level, and can directly influence downstream semantic tasks. It bridges the gap between task-based explanation and vector-space analysis of existing studies on embedding interpretability, and offers a systematic approach for understanding and tuning embeddings in real-world applications.

References

D. Achlioptas. Database-friendly random projections: Johnson–Lindenstrauss with binary coins. *Jour. of Computer and System Sciences*, 66(4):671–687, 2003. doi: 10.1016/S0022-0000(03)00025-4.

E. Bingham and H. Mannila. Random projection in dimensionality reduction: Applications to image and text data. In *Proc. ACM SIGKDD Int.Conf. on Knowledge Discovery and Data Mining*, pages 245–250, 2001. doi: 10.1145/502512.502546.

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Jour. of Machine Learning Research*, 3:993–1022, 2003. doi: 10.1162/jmlr.2003.3.4-5.993.

Y. Bu, M. Li, W. Gu, and W. Huang. Topic diversity: A discipline scheme-free diversity measurement for journals. *Jour. of the American Society for Information Science and Technology*, 72(5):523–539, 2021. doi: 10.1002/asi.24433.

X. Cheng, X. Yan, Y. Lan, and J. Guo. Btm: Topic modeling over short texts. *IEEE Trans.on Knowledge and Data Engineering*, 26(12):2928–2941, 2014. doi: 10.1109/TKDE.2014.2313872.

R. Das, M. Zaheer, and C. Dyer. Gaussian LDA for topic models with word embeddings. In *Proc. Annual Meeting of the Association for Computational Linguistics and Int. Joint Conf. on Natural Language Processing*, pages 795–804, 2015. Association for Computational Linguistics. doi: 10.3115/v1/P15-1077.

S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Jour. of the American Society for Information Science and Technology*, 41(6):391–407, 1990. doi: 10.1002/(SICI)1097-4571(199009)41:6⟨391::AID-ASI1⟩3.0.CO;2-9.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423.

K. Ethayarajh. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proc. Conf. on Empirical Methods in Natural Language Processing*, pages 55–65, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1006.

J. Ganesh, G. Manish, and V. Vasudeva. Doc2sent2vec: A novel two-phase approach for learning document representation. In *Proc. ACM-SIGIR Int. Conf. on Research and Development in Information Retrieval*, page 809–812, 2016. Association for Computing Machinery. doi: 10.1145/2911451.2914717.

D. Ganguly, D. Roy, M. Mitra, and G. J. F. Jones. Word embedding based generalized language model for information retrieval. In *Proc. ACM-SIGIR Int. Conf. on Research and Development in Information Retrieval*, SIGIR '15, page 795–798, 2015. Association for Computing Machinery. doi: 10.1145/2766462.2767780.

N. Gillis and S. A. Vavasis. Fast and robust recursive algorithmsfor separable nonnegative matrix factorization. *IEEE Trans.on Pattern Analysis and Machine Intelligence*, 36(4): 698–714, 2014. doi: 10.1109/TPAMI.2013.226.

M. Grootendorst. Bertopic: Neural topic modeling with a class-based TF-IDF procedure. *ArXiv*, abs/2203.05794, 2022. doi: 10.48550/arXiv.2203.05794.

J. Guo, Y. Cai, Y. Fan, F. Sun, R. Zhang, and X. Cheng. Semantic models for the first-stage retrieval: A comprehensive review. *ACM Trans.on Information Systems*, 40(4), 2022. doi: 10.1145/3486250.

H. Gupta and M. Patel. Method of text summarization using LSA and sentence based topic modelling with BERT. In *Int. Conf. on Artificial Intelligence and Smart Systems*, pages 511–517, 2021. doi: 10.1109/ICAIS50930.2021.9395976.

S. Han, M. Shin, S. Park, C. Jung, and M. Cha. Unified neural topic model via contrastive learning and term weighting. In *Proc. Conf. European Chapter of the Association for Computational Linguistics*, pages 1802–1817, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.132.

H. A. M. Hassan, G. Sansonetti, F. Gasparetti, and A. Micarelli. Semantic-based tag recommendation in scientific bookmarking systems. In *Proc. ACM Conf. on Recommender Systems*, page 465–469, 2018. Association for Computing Machinery. doi: 10.1145/3240323.3240409.

F. Incitti, F. Urli, and L. Snidaro. Beyond word embeddings: A survey. *Information Fusion*, 89:418–436, 2023. ISSN 1566-2535. doi: 10.1016/j.inffus.2022.08.024.

A. Jha, V. Rakesh, J. Chandrashekar, A. Samavedhi, and C. K. Reddy. Supervised contrastive learning for interpretable long-form document matching. *ACM Trans.on Knowledge Discovery from Data*, 17(2), 2023. doi: 10.1145/3542822.

Y. Jin, H. Zhao, M. Liu, L. Du, and W. Buntine. Neural attention-aware hierarchical topic model. In *Proc. Conf. on Empirical Methods in Natural Language Processing*, pages 1042–1052, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.80.

W. B. Johnson and J. Lindenstrauss. Extensions of lipschitz mappings into a hilbert space. In *Conf. in Modern Analysis and Probability*, volume 26 of *Contemporary Mathematics*, pages 189–206. American Mathematical Society, 1984. doi: 10.1090/conm/026/737400.

P. Karvelis, D. Gavrilis, G. Georgoulas, and C. Stylios. Topic recommendation using Doc2Vec. In *Int. Joint Conf. on Neural Networks*, pages 1–6, 2018. IEEE. doi: 10.1109/IJCNN.2018.8489513.

H. K. Kim, H. Kim, and S. Cho. Bag-of-concepts: Comprehending document representation through clustering words in distributed representation. *Neurocomputing*, 266:336–352, 2017. doi: 10.1016/j.neucom.2017.05.046.

K. Lang. 20 newsgroups dataset, 1995. URL `http://people.csail.mit.edu/jrennie/20Newsgroups/`.

Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *Proc. Int. Conf. on Machine Learning*, pages 1188–1196, 2014. JMLR.org.

X. Ma, L. Wang, N. Yang, F. Wei, and J. Lin. Fine-tuning llama for multi-stage text retrieval. In *Proc. ACM-SIGIR Int. Conf. on Research and Development in Information Retrieval*, SIGIR '24, page 2421–2425, 2024. Association for Computing Machinery. doi: 10.1145/3626772.3657951.

A. Meddeb and L. B. Romdhane. Using topic modeling and word embedding for topic extraction in Twitter. *Procedia Computer Science*, 207(C):790–799, 2022. ISSN 1877-0509. doi: 10.1016/j.procs.2022.09.134.

Y. Meng, Y. Zhang, J. Huang, Y. Zhang, and J. Han. Topic discovery via latent space clustering of pretrained language model representations. In *Proc. World-Wide Web Conference*, page 3143–3152, 2022. Association for Computing Machinery. doi: 10.1145/3485447.3512034.

T. Mikolov, K. Chen, G. S. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *ArXiv*, abs/1301.3781, 2013. doi: 10.48550/arXiv.1301.3781.

Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019. doi: 10.1016/j.artint.2018.07.007.

D. Mimno, H. M. Wallach, E. Talley, M. Leenders, and A. McCallum. Optimizing semantic coherence in topic models. In *Proc. Conf. on Empirical Methods in Natural Language Processing*, page 262–272, 2011. Association for Computational Linguistics. doi: 10.5555/2145432.2145462.

F. Morstatter and H. Liu. In search of coherence and consensus: Measuring the interpretability of statistical topics. *Jour. of Machine Learning Research*, 18(1):6177–6208, 2017. ISSN 1532-4435. doi: 10.5555/3122009.3242026.

W. J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, and B. Yu. Definitions, methods, and applications in interpretable machine learning. *Proc. National Academy of Sciences*, 116(44):22071–22080, 2019. doi: 10.1073/pnas.1900654116.

B. A. H. Murshed, S. Mallappa, J. Abawajy, M. A. N. Saif, H. D. E. Al-ariki, and H. M. Abdulwahab. Short text topic modelling approaches in the context of big data: taxonomy, survey, and analysis. *Artificial Intelligence Review*, 56(6):5133–5260, 2022. ISSN 0269-2821. doi: 10.1007/s10462-022-10254-w.

M. Nauta, J. Trienes, S. Pathak, E. Nguyen, M. Peters, Y. Schmitt, Y. Schlötterer, M. van Keulen, and C. Seifert. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI. *ACM Computing Surveys*, 55(13s), 2023. ISSN 0360-0300. doi: 10.1145/3583558.

D. Newman, J. H. Lau, K. Grieser, and T. Baldwin. Automatic evaluation of topic coherence. In *Annual Conf. North American chapter of the association for computational linguistics*, pages 100–108, 2010. Association for Computational Linguistics. doi: 10.5555/1857999.1858011.

A. Panigrahi, H. V. Simhadri, and C. Bhattacharyya. Word2Sense: Sparse interpretable word embeddings. In *Proc. Annual Meeting of the Association for Computational Linguistics*, pages 5692–5705, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1570.

J. Pennington, R. Socher, and C. Manning. GloVe: Global vectors for word representation. In *Proc. Conf. on Empirical Methods in Natural Language Processing*, pages 1532–1543, 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162.

M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. In *Annual Conf. North American chapter of the association for computational linguistics*, pages 2227–2237, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1202.

T. Prouteau, N. Dugué, N. Camelin, and S. Meignier. Are embedding spaces interpretable? results of an intrusion detection evaluation on a large French corpus. In *Proc. Language Resources and Evaluation Conference*, pages 4414–4419, 2022. European Language Resources Association.

N. Reimers and I. Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proc. Conf. on Empirical Methods in Natural Language Processing and Int. Joint Conf. on Natural Language Processing*, pages 3982–3992, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410.

S. Seifollahi, M. Piccardi, and A. Jolfaei. An embedding-based topic model for document classification. *ACM Trans.Asian Low-Resources Language Information Processing*, 20(3), 2021. doi: 10.1145/3431728.

L. K. Şenel, F. Şahinuç, V. Yücesoy, H. Schütze, T. Çukur, and A. Koç. Learning interpretable word embeddings via bidirectional alignment of dimensions with semantic concepts. *Information Processing & Management*, 59(3):102925, 2022. doi: 10.1016/j.ipm.2022.102925.

S. Sia, A. Dalmia, and S. J. Mielke. Tired of topic models? clusters of pretrained word embeddings make for fast and good topics too! In *Proc. Conf. on Empirical Methods in Natural Language Processing*, pages 1728–1736, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.135.

C. Singh, A. Askari, R. Caruana, and J. Gao. Augmenting interpretable models with large language models during training. *Nature Communications*, 14(1):7913, 2023. doi: 10.1038/s41467-023-43713-1.

J. Steinberger and K. Ježek. Text summarization and singular value decomposition. In Tatyana Yakhno, editor, *Advances in Information Systems*, pages 245–254, 2005. Springer Berlin Heidelberg.

A. Subramanian, D. Pruthi, H. Jhamtani, T. Berg-Kirkpatrick, and E. H. Hovy. Spine: Sparse interpretable neural embeddings. *ArXiv*, abs/1711.08792, 2017. doi: 10.48550/arXiv.1711.08792.

S. Tanabe, M. Ohta, A. Takasu, and J. Adachi. An approach to estimating cited sentences in academic papers using Doc2Vec. In *Proc. Int. Conf. on Management of Digital EcoSystems*, page 118–125, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356220. doi: 10.1145/3281375.3281391.

J. Tao, L. Zhou, and K. Hickey. Making sense of the black-boxes: Toward interpretable text classification using deep learning models. *Jour. of the American Society for Information Science and Technology*, 74(6):685–700, 2023. doi: 10.1002/asi.24642.

F. Viegas, S. Canuto, C. Gomes, W. Luiz, T. Rosa, S. Ribas, L. Rocha, and M. A. Gonçalves. Cluwords: exploiting semantic word clustering representation for enhanced topic modeling. In *Proc. ACM Int. Conf. on Web Search and Data Mining*, pages 753–761, 2019. Association for Computing Machinery. doi: 10.1145/3289600.3291032.

E. M. Voorhees and D. K. Harman, editors. *TREC Experiment and Evaluation in Information Retrieval*. The MIT Press, 2005. ISBN 9780262220736.

I. Vulić and M. Moens. Monolingual and cross-lingual information retrieval models based on (bilingual) word embeddings. In *Proc. ACM-SIGIR Int. Conf. on Research and Development in Information Retrieval*, page 363–372, 2015. Association for Computing Machinery. doi: 10.1145/2766462.2767752.

Y. Wang and Y. Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Trans.on Knowledge and Data Engineering*, 25(6):1336–1353, 2012. doi: 10.1109/TKDE.2012.51.

C. Wu, E. Kanoulas, and M. Rijke. Learning entity-centric document representations using an entity facet topic model. *Information Processing & Management*, 57(3):102216, 2020. ISSN 0306-4573. doi: 10.1016/j.ipm.2020.102216.

X. Yang, D. Lo, X. Xia, L. Bao, and J. Sun. Combining word embedding with information retrieval to recommend similar bug reports. In *IEEE Int. Symposium on Software Reliability Engineering*, pages 127–137, New York, NY, United States, 2016. IEEE. doi: 10.1109/ISSRE.2016.33.

M. Yuan, P. Lin, L. Rashidi, and J. Zobel. Asessment of the quality of topic models for information retrieval applications. In *Proc. ACM-SIGIR Int. Conf. on Theory of Information Retrieval*, ICTIR '23, 2023. Association for Computing Machinery. doi: 10.1145/3578337.3605118.

J. Zhan, J. Mao, Y. Liu, M. Zhang, and S. Ma. Repbert: Contextualized text embeddings for first-stage retrieval. *ArXiv*, abs/2006.15498, 2020. doi: 10.48550/arXiv.2006.15498.

T. Zheng and M. Wang. Using SVD for topic modeling. *Jour.of the American Statistical Association*, 0(0):1–16, 2022. doi: 10.1080/01621459.2022.2123813.

G. Zuccon, B. Koopman, P. Bruza, and L. Azzopardi. Integrating and evaluating neural word embeddings in information retrieval. In *Proc. Australian Document Computing Conf.*, 2015. Association for Computing Machinery. doi: 10.1145/2838931.2838936.

Y. Zuo, C. Li, H. Lin, and J. Wu. Topic modeling of short texts: A pseudo-document view with word embedding enhancement. *IEEE Trans.on Knowledge & Data Engineering*, 35 (01):972–985, 2023. ISSN 1558-2191. doi: 10.1109/TKDE.2021.3073195.

Appendix A. Qualitative Mapping Performance Tables

In this Appendix we show examples of mapped and unmapped topics to illustrate the quantitative results in the body of the paper. Although strong conclusions cannot be drawn from these illlustrations, intuitively the mapped topics tend to be more interpretable (or semantically consistent) than the unmapped – noting that we have included cases even though they are counterexamples, such as topic 60 in Table 4 and topic 19 in Table 5.

Embedding	TopicID	#Feat.	Keyword Description
Doc2Vec	2	9	think, go, like, know, time, get, want, thing, come, way
	22	7	god, jesus, church, christ, sin, love, christian, bible, lord, say
	0	6	space, earth, planet, moon, launch, solar, orbit, spacecraft, system, mission
Sentence-BERT	52	8	price, sell, new, sale, offer, good, buy, include, interested, like
	62	7	game, play, period, goal, score, shot, lead, pt, espn, win
	22	6	god, jesus, church, christ, sin, love, christian, bible, lord, say
RepLLaMa	2	21	think, go, like, know, time, get, want, thing, come, way
	45	18	armenian, turkish, turk, turkey, russian, armenians, genocide, muslim, population, village
	15	8	window, problem, thank, help, run, work, know, try, use, like
Unmapped	24		orthodox, son, slipper, till, candida, italy, beast, abstract, explore, presentation
	68		keyboard, sub, rgb, virtual, thanx, custom, interior, silicon, plastic, fish
	54		dry, push, playback, evolution, depth, clean, pop, educational, reverse, quit

Table 2: Mapped and unmapped topics for the 20NG collection with LDA. The 'Feat.' column is the number of features matched to the corresponding topic. The 'Unmapped' row is three unmapped topics.

Embedding	TopicID	#Feat.	Keyword Description
Doc2Vec	8	13	smiley, manhattan, beauchaine say, blow bronx, manhattan sea, bronx away, queen stay, say queen, stay blow, sink manhattan
	4	8	gun, homosexual, people, right, gay, drug, sex, moral, think, man
	2	6	car, price, new, bike, sell, sale, engine, good, appear, buy
Sentence-BERT	8	7	smiley, manhattan, beauchaine say, blow bronx, manhattan sea, bronx away, queen stay, say queen, stay blow, sink manhattan
	2	7	car, price, new, bike, sell, sale, engine, good, appear, buy
	1	6	key, encryption, chip, clipper, government, use, message, algorithm, phone, system
RepLLaMa	0	8	game, team, play, player, win, year, season, hockey, league, hit
	30	7	battery, discharge, concrete, charge, acid, temperature, heat, lead, reaction, dirt
	1	6	key, encryption, chip, clipper, government, use, message, algorithm, phone, system
Unmapped	35		donation, copyright, shareware, john, notice, author, commercial, copy, fee, jan
	29		helmet, pocket, jacket, piece, liner, fit, pant, woman, clothing, foam
	28		mouse, driver, ball, problem, apple, mouse driver, window, roller, load, button

Table 3: Mapped and unmapped topics for the 20NG collection with BERTopic. The 'Feat.' column is the number of features matched to the corresponding topic. The 'Unmapped' row is three unmapped topics.

Embedding	TopicID	#Feat.	Keyword Description
Doc2Vec	24	8	empire, roman, emperor, war, city, army, king, battle, byzantine, rome
	14	8	philosophy, theory, knowledge, view, philosopher, school, idea, truth, nature, world
	43	6	oil, water, gas, fuel, air, increase, fire, energy, carbon, chemical
Sentence-BERT	12	5	roman, rome, empire, italy, emperor, latin, romans, greek, italian, city
	66	5	university, publish, society, science, laplace, mathematic, research, write, book, professor
	16	5	culture, cultural, people, group, country, right, human, united, social, community
RepLLaMa	45	8	language, italian, dialect, speak, latin, romance, french, vowel, word, italy
	25	6	integral, function, theory, series, theorem, set, problem, calculus, mathematic, mathematical
	24	6	empire, roman, emperor, war, city, army, king, battle, byzantine, rome
Unmapped	60		bridge, arch, build, river, stone, aqueduct, span, construct, roman, welding
	76		madhava, motto, sine, trigonometric, series, school, seal, magister, arc, acta
	48		instrument, curl, cosmetic, musical, rope, perfume, hair, hand, scallop, ensemble

Table 4: Mapped and unmapped topics for the WIKI collection with LDA. The 'Feat.' column is the number of features matched to the corresponding topic. The 'Unmapped' row is three unmapped topics.

Embedding	TopicID	#Feat.	Keywords
Doc2Vec	0	12	roman, rome, emperor, empire, city, caesar, century, senate, romans, military
	30	5	plants, plant, species, taxonomy, glossary, organisms, flora, animals, biology, botany
	3	4	italian, sardinian, language, dialect, catalan, speak, italy, calabria, occitan, romance
Sentence-BERT	0	14	roman, rome, emperor, empire, city, caesar, century, senate, romans, military
	4	7	greek, the, of, was, and, were, in, empire, greeks, to
	2	5	letter, letters, vowel, alphabet, syllable, used, greek, style, form, character
RepLLaMa	19	10	displaystyle, x2212, x2061, integral, x222b, int, function, x221e, x03c0, integrals
	0	8	roman, rome, emperor, empire, city, caesar, century, senate, romans, military
	47	8	shen, chinese, china, dynasty, confucian, confucius, han, mao, dynasty, ruler
Unmapped	56		milk, feed, livestock, dairy, animal, production, farm, feed, cow, bee
	73		babylon, assyrian, city, assyria, king, makuria, baghdad, bc, antioch, mesopotamia
	55		oil, fuel, bitumen, gasoline, sands, diesel, petroleum, asphalt, crude, refinery

Table 5: Mapped and unmapped topics for the WIKI collection with BERTopic. The 'Feat.' column is the number of features matched to the corresponding topic. The 'Unmapped' row is three unmapped topics.

Embedding	TopicID	#Feat.	Keywords
Doc2Vec	48	6	business, market, big, industry, time, go, small, cost, analyst, buy
	47	5	bush, party, political, campaign, election, democratic, president, state, candidate, vote
	40	4	like, people, time, know, day, go, come, get, man, old
Sentence-BERT	28	8	president, executive, chief, vice, officer, chairman, name, director, old, board
	24	6	share, stock, common, exchange, cent, trading, close, outstanding, inc, dividend
	46	4	drug, health, medical, care, hospital, research, patient, fda, test, study
RepLLaMa	39	12	sell, agreement, group, inc, unit, corp, sale, agree, acquire, transaction
	24	6	share, stock, common, exchange, cent, trading, close, outstanding, inc, dividend
	28	6	president, executive, chief, vice, officer, chairman, name, director, old, board
Unmapped	13		food, restaurant, franchise, chain, india, fast, spanish, taylor, knight, rice
	33		san, pacific, california, francisco, southern, santa, railroad, georgia, diego, forest
	43		insurance, steel, life, insurer, cellular, premium, policy, national, corp, industry

Table 6: Mapped and unmapped topics for the WSJ collection with LDA. The 'Feat.' column is the number of features matched to the corresponding topic. The 'Unmapped' row is three unmapped topics.

Embedding	TopicID	#Feat.	Keywords
Doc2Vec	0	6	share, say, company, million, stock, year, market, inc, new, sale
	1	5	bonds, bank, yield, treasury, rate, dollar, issue, million, price, market
	3	4	rise, increase, rate, adjust, month, year, unemployment, economist, consumer, price
Sentence-BERT	0	17	share, say, company, million, stock, year, market, inc, new, sale
	3	6	rise, increase, rate, adjust, month, year, unemployment, economist, consumer, price
	23	4	kodak, polaroid, cannon, shamrock, film, camera, tape, recorder, say, video
RepLLaMa	0	26	share, say, company, million, stock, year, market, inc, new, sale
	46	4	islam, moslem, koran, ayatollah, mosque, iran, khomeini, indonesia, religion, islamic
	1	3	bonds, bank, yield, treasury, rate, dollar, issue, million, price, market
Unmapped	38		constitution, convention, church, woman, constitutional, madison, amendment, people, document, hofmann
	31		pri, salinas, mexico, election, pinochet, party, opposition, mexican, gen, haiti
	28		south, africa, black, african, apartheid, anc, south african, government, white, south africa

Table 7: Mapped and unmapped topics for the WSJ collection with BERTopic. The 'Feat.' column is the number of features matched to the corresponding topic. The 'Unmapped' row is three unmapped topics.