

Optimizing Neural Topic Modeling Pipelines for Low-quality Speech Transcriptions

Ehsan Taati, Marcin Budka, Scott Neville, and James Canniffe

¹ Bournemouth University

² Bluestar Software Ltd

Abstract. Gaining insights from large-scale document archive is a challenging task. Recent advances in natural language processing, specifically unsupervised topic modeling, allow for automated discovery of abstract “topics” that characterize groups of semantically related documents within textual corpora. Neural topic modeling has emerged as a scalable approach through integrating state-of-the-art sentence embedding models into modeling pipelines. This embedding-based architecture enables efficient processing of large datasets. However, topic quality often related to input data quality, particularly in the case of speech-to-text, remains an open issue. This study presents a comparative evaluation of various component configurations within a neural topic modeling pipeline, as applied to a corpus of telephony transcriptions. Incorporating four embedding models (E5, Instructor, MiniLM, and SGPT), three dimensionality reduction approaches (maintaining versus reducing original embeddings by Truncated-SVD and UMAP), and two clustering algorithms (K-Means and HDBSCAN), 48 topic modeling pipelines are evaluated. The experimental results reveal that placing a context-aware embedding model in the pipeline leads to significant improvement in topic coherence, while larger models tend to achieve better topic diversity. Based on the above, we also propose best practices of the model layout in the pipeline, considering coherence and topic diversity scores.

Keywords: Topic modeling · Embedding Model · Large-Scale Text.

1 Introduction

Topic modeling refers to a family of approaches for adding a high level of abstraction to a large archive of documents by extracting the latent variables known as topics, typically utilizing unsupervised machine learning. Topic representations can be interpreted by humans to categorize the archived documents for further use [2]. Amongst approaches to topic modeling, neural models have grown in popularity in recent years due to their flexibility and scalability. Neural topic modeling architectures fundamentally comprise an embedding model for text embedding coupled with a clustering module for discovering latent topics based on the embedded document representations. The embedding model leverages deep neural networks already trained on massive text corpora to encode input

documents into dense numerical representations that preserve linguistic meaning. This allows neural approaches to ingest large-scale, domain-diverse corpora while traditional methods struggle with vocabulary variation. The subsequent clustering step partitions the embedded documents based on distributional similarity, with each cluster corresponding to a distinct topic [8, 19]. Utilizing different embedding models in the topic modeling pipeline even on the same document corpus may lead to different outcomes. Properties of embedding models including network architectural complexity, scale of pre-training as measured by model parameters, and specialization for supervised or unsupervised end tasks have proven critical for performance across downstream tasks [16]. Similarly, research has demonstrated choice of dimensionality reduction technique and clustering algorithm significantly impacts neural topic modeling effectiveness [5, 25]. Moreover, repeating topic modeling on the same data with identical parameter settings does not guarantee consistent results due to stochastic nature of some of the unsupervised algorithms used in the pipeline [10].

Although topic modeling has been studied in various fields such as bioinformatics and ecology, the main application of it lies in textual data analysis [25]. A text archive may contain a variety of written texts such as news articles, scientific papers, or transcriptions from multimedia sources including social media, meetings, and telephony conversations. Transcriptions are commonly generated by Automatic Speech Recognition (ASR) models from audio data sources. Although recent advances in deep learning revolutionized the ASR models, they are still negatively influenced by the environmental conditions such as background noise and reverberation, and speakers’ characteristics such as tiredness or accents [6]. The manifestation of all these factors occurs in telephony speech, with additional degradation due to transmission through telephone network [11].

The above limitations in both modeling and data aspects, make topic modeling a particularly challenging task in real-world setting, especially when applied to a telephony conversation corpus. Hence, we examine the effect of incorporating different components in a topic modeling pipelines with an additional emphasis on the role of embedding models on a large-scale real-world dialogue-based transcription corpus. To this end, we systematically investigate 48 neural topic modeling pipelines resulting from all permutations of 4 embedding models, 3 dimensionality reduction techniques, and 2 clustering algorithms. Our experiments utilize the recent BERTopic [8] framework for neural topic modeling to enable standardized comparison across pipeline configurations. Additionally, optimized configurations including the embedding model, dimensionality reduction, and clustering algorithm are identified across pipelines to establish best practice recommendations for balancing trade-offs in topic coherence and diversity.

2 Related Work

Current research on topic modeling is highly focused on textual data in the form of written documents. However, considering the availability of data in the form of long transcriptions derived from audio sources, there is a need to expand the

scope of topic modeling beyond human-authored textual data. Here we focus on the limited number of existing studies that employ topic modeling on lengthy texts that are artificially generated. To gain a deeper understanding of this niche area of research, the related existing works are categorized into two main groups: (1) spoken non-conversational, and (2) spoken conversational texts.

Arazzi et al.[1] employ BERTopic to analyze recordings of video lessons that have been transcribed using an ASR. They filter out conversational sections to improve topic quality. Their pipeline follows the standard BERTopic methodology with no customizations. A qualitative analysis is incorporated to evaluate model performance. Another study by [24] applies topic modeling to detect polarization and toxicity in TikTok video transcriptions generated using Whisper [20]. Whisper is a noise- and accent-robust transformer trained on large speech corpora. The preprocessed texts are analyzed using a topic model tailored for short texts to extract themes. Despite negative coherence scores, their semantic analysis expresses adequate model performance. Tshimula [23] recently characterized the financial market by topic modeling YouTube videos related to financial news, with transcriptions produced with Whisper. Documents undergo extensive normalization prior to model fitting. Comparative assessments prefer BERTopic over classical approaches for its ability to process complex corpora.

Papadia et al.[19] compare four non-neural topic modeling algorithms, including Latent Dirichlet allocation (LDA) [3], on phone conversation transcriptions. Optimal outcomes using LDA with 20 topics were achieved according to coherence (0.63), topic diversity (0.18) and classification score (0.39) metrics. However, crucial details including dataset size, structure, labeling process for classification, and embedding techniques are not disclosed. Khalid et al. [9] present a comparative analysis of topic detection via conversational text by combining clustering with an LDA variant in a semi-supervised framework applied to three academic dialog datasets. Their model achieves higher accuracy on partially labeled data (0.866, 0.843, 0.764) versus the unsupervised version. Pan et al. [18] formulate topic modeling as a sequence labeling task on customer service transcriptions with predefined topics. Sentences are the input element based on dialog text characteristics. Sliding windows sample inputs using the embedding model’s maximum sequence length. A fully connected layer then predicts topic label sequences. Multiple BERT-based models are trained using k-fold cross validation to improve generalization and robustness. A 3-model ensemble with voting surpasses single models by 2.03% in F1.

3 Methods

This study employs BERTopic [8] to extract latent topics from transcriptions of phone conversations recorded in a call center. BERTopic is an unsupervised model comprising four steps to identify and extract topics. First, a sentence embedding model transforms input documents into vector representations. Next, an optional dimensionality reduction step reduces the high-dimensional embedding space to prepare for clustering. In the third step, a clustering algorithm par-

titions the embedded documents by grouping distributionally similar vectors. Finally, the Term-Frequency/Inverse Document Frequency (TF-IDF) extracts topic-words from each merged document cluster as topic representations. Each module can leverage alternative methods or algorithms given the objective and data characteristics. To determine optimal configurations, 48 experiments systematically combine 2 data processing strategies, 4 embedding models, 3 dimensionality reduction methods, and 2 clustering algorithms.

3.1 Embedding Models

Embedding models play a crucial role in the BERTopic architecture by embedding input text into dense vectors for clustering. The embedding model establishes relationships between sentences in a document [28]. Four embedding models were selected by consulting the Massive Text Embedding Benchmark leaderboard³ based on two criteria associated with the task and data: clustering score and maximum input sequence length (Table 1). MiniLM [27] is a distilled

Table 1: Embedding Models Information

Embedding Model	Size (GB)	Embeddings Dimensions	Max Sequence Length	Clustering Score
All-MiniLM-L6-v2	0.09	384	256	42.35
E5-Large-V2	1.34	1024	512	44.49
Instructor-XL	4.96	768	512	44.74
SGPT	23.5	4096	2048	40.34

embedding model that reduces the 109 million parameters of the original BERT model down to 66 million while maintaining performance. The authors employ knowledge distillation to compress a large ‘teacher’ model into a smaller ‘student’ model with substantially fewer parameters that can still complete downstream tasks. MiniLM training is based on three core concepts: 1) mimicking the self-attention module, 2) transferring the query-key-value relational structure from teacher to student, and 3) distilling the student via an intermediate ‘assistant teacher’ model.

Embeddings from Bidirectional Encoder Representations (E5) [26] is an embedding model trained using a contrastive learning approach, where the model is presented with similar and dissimilar text examples to improve on predecessors like BERT. The authors curated an unlabeled 1.3 billion training text pair dataset by combining multiple semi-structured data sources for contrastive pre-training. The E5 transformer-based pipeline comprises two stages: contrastive pre-training and fine-tuning.

To mitigate performance degradation when applying text embeddings to new tasks or domains, Su et al. [22] introduced Instruction-based Omnifarious Representations (Instructor), a single multitask embedding model producing task- and

³ <https://huggingface.co/spaces/mteb/leaderboard>

domain-aware embeddings via instruction-based fine-tuning. The instructions can specify the text type (sentence, paragraph, document), task (information retrieval, clustering), and domain (News, music, sport, etc.).

GPT Sentence Embeddings for Semantic Search (SGPT) [15] utilizes a 5.8 billion parameter decoder-only transformer to improve performance and reduce computation via reusing the large decoder for diverse tasks. The goal is to leverage the scaling of GPT-like encoder-only transformers within sentence embedding models. SGPT introduces two sentence embedding structures: a Cross-Encoder for semantic search encoding query-document concatenations, and a Bi-Encoder encoding queries and documents separately to capture semantics.

3.2 Dimensionality Reduction Methods

Dimensionality reduction (DR) methods reduce input feature dimensionality while preserving important information and relationships for subsequent tasks. Linear dimensionality reductions assume linear feature relationships and provide significant dimension reduction. Non-linear methods are preferred for complex non-linear data [17]. This study investigates the effect of dimensionality reduction on topic modeling using one linear method (Truncated SVD), one non-linear method (UMAP), and no dimensionality reduction.

Singular Value Decomposition (SVD) and its variant truncated SVD (Truncated-SVD) mitigate the negative impact of outliers in text via matrix factorization into orthogonal and diagonal singular value matrices. By preserving only the top k singular values and vectors, truncated SVD retains crucial information while disregarding less important details [7].

Uniform Manifold Approximation and Projection (UMAP) is a recent manifold learning technique for dimensionality reduction addressing scalability and diversity issues of its predecessors. While primarily intended to improve data visualization via preserving global structure, UMAP has become widely used in machine learning due to imposing no constraints on output dimensions. It constructs a weighted k-neighbor graph, then computes a low dimensional layout while focusing on maintaining topological representations [14].

3.3 Clustering Methods

The choice of clustering algorithm impacts the quantity and quality of extracted topics. Bora et al. [4] categorize clustering methods as either soft, allowing data points to have partial membership in multiple clusters, or hard, with definite cluster assignments. To investigate the impact of soft versus hard clustering, we compare two prevalent algorithms: K-means (hard) and HDBSCAN (soft).

K-Means [12] requires pre-defining the number of clusters k to partition data points into k groups. The algorithm aims to maximize intra-cluster similarity and minimize inter-cluster similarity. Random initialization of clusters can influence results, thus multiple K-Means runs helps mitigate this limitation.

Traditional clustering algorithms suffer from limitations like requiring pre-defined cluster counts, or distributional assumptions of the clusters. Hierarchi-

cal Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) addresses these issues as a soft clustering approach via constructing a cluster hierarchy and tree based on density reachability between data points, allowing clusters to emerge naturally based on intrinsic data density properties without imposing positional or distributional restrictions. Moreover, HDBSCAN facilitates identification of noise by labeling any data points not assigned to a cluster as potential outliers for further analysis [13].

3.4 Topic Representation

To represent clusters as topics, BERTopic uses a modified TF-IDF approach. All documents in a cluster are concatenated into one mega-document on which TF-IDF scores are calculated, reflecting word importance for the cluster rather than individual documents. An iterative process then merges the least significant terms with their most similar representations. This assigns an exclusive topic to each cluster based on its constituent documents [8].

3.5 Data

This study utilizes 6,770 hours of UK police emergency call center recordings, transcribed with Whisper. We employed two pre-processing strategies to produce corpora with markedly different document length distributions.

The Whisper transcriptions contained non-trivial errors negatively impacting topic quality. Examples include non-ASCII characters, greedy decoding causing word/sentence repetition in due to the presence of background noise in recorded conversations, and tone-related terms like “beep” and “dot”. To mitigate these deficiencies, an *improved corpus* was produced by identifying and resolving transcription errors via tailored regular expressions. Matching patterns removed or corrected the aforementioned issues while minimizing text alterations to maintain comparability to the original transcriptions.

Given existing works indicating extensive pre-processing benefits neural topic models, a *normalized corpus* was created by fixing contractions common in dialog, removing digits and punctuation, applying stemming, and lowercasing.

Due to embedding models truncating inputs surpassing their maximum sequence lengths (Table 1), documents were segmented to embed full content when exceeding the threshold. Segmentation can substantially increase corpus size for models with smaller limits. In total, eight datasets resulted from pre-processing such that each embedding model was applied to two corpora.

3.6 Evaluation Metrics

Topic coherence and topic diversity are widely used metrics for evaluating topic models [8, 10, 19]. However, human judgment of topic quality is still invaluable in practice [10]. Topic coherence is measured using Normalized Pointwise Mutual Information (NPMI) scores, calculating the statistical co-occurrence between

topic words on a scale of -1 to 1 (Eq. 1). An NPMI of 1 denotes maximum coherence. Röder et al. demonstrated strong correlation between NPMI values and human judgments of topic coherence [21].

$$C_{NPMI} = (P_{sw}(10), S_{one}^{one}, m_{nlr}, \sigma_{\alpha}) \quad (1)$$

where $P_{sw}(10)$ indicates word counting using sliding window of 10, S_{one}^{one} is a confirmation measure for single pair of words or subset of words, m_{nlr} identifies normalized log-ratio measure, and σ_{α} represents arithmetic mean.

Topic Diversity (TD) measures overlap between topics, defined as the percentage of unique words across all topic representations. TD scores range from 0 to 1, with values nearing 0 indicating high redundancy between topics and 1 denoting completely distinct topics with no overlap in content [5].

3.7 Experiment Setup

Systematically permuting datasets, dimensionality reduction approaches, clustering algorithms, and embedding models resulted in 48 distinct topic modeling pipelines. In alignment with recommended practices for equitable benchmarking, each architecture underwent fitting across a range of topic numbers from 10 to 50 in steps of 10. To further mitigate output instability, each experiment was repeated 3 times. The topic modeling pipelines were implemented with the BERTopic framework⁴. Model evaluation leveraged the OCTIS⁵ platform. To instruct the Instructor-XL model, the prompt *Represent the document for clustering* was utilized. This directed the model to generate embeddings tailored to the document-level text and optimized for the clustering task.

4 Results

This study examined combinations of topic modeling pipeline components, including 4 embedding models shown effective for unsupervised tasks, 3 dimensionality reduction, and 2 clustering algorithms applied to 2 corpora of large-scale call transcriptions.

Topic modeling performance was compared between the two text pre-processing strategies, *normalized* versus *improved corpus*, aggregated across all pipeline component permutations, as shown in Table 2. It is clear that *normalized* text pre-processing improves coherence (0.0449) and topic diversity (0.7447). While neural embedding models can theoretically ingest documents and generate representations from non-alphanumeric characters, thorough pre-processing to remove such aberrations improves topic coherence. These normalized embeddings contain increased semantic content, better equipped for subsequent steps. Comprehensive cleaning enables more meaningful textual data to be embedded, producing enriched document vectors compared to original error-filled transcripts. This is aligned with the top performing components layout utilizing the *normalized corpus* Table 6.

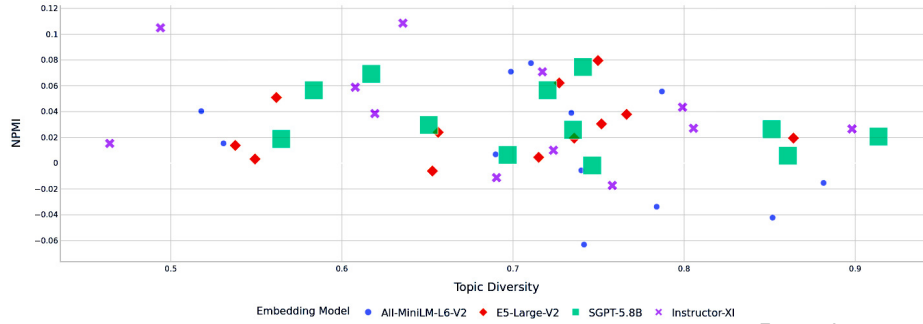


Fig. 1: Performance trade-off with respect to average topic coherence (NPMI) and topic diversity for embedding models (shape size = max. sequence length)

Table 2: Effect of text pre-processing on average performance

Corpus	NPMI	TD
Normalized corpus	0.0449	0.7447
Improved Corpus	0.0113	0.6649

Table 3 summarizes embedding models’ effects in the experiment. Instructor-XL achieved highest coherence (0.0396) followed by SGPT5.8B (0.0324). The Instructor-XL embeddings were generated by prompting the model with explicit instructions specifying the details of the downstream clustering task., explained in 3.7. Results indicate context-aware models improve topic coherence regardless of model size, sequence length, or embedding dimensions. SGPT5.8B possesses the most parameters and longest sequences while E5-Large creates highest-dimensional embeddings, yet underperformed Instructor-XL. All-MiniLM-L6-V2 exhibited poor coherence. However, SGPT5.8B achieved best diversity (0.7234 TD), slightly exceeding the smallest model (0.7223 TD). Text segmentation increasing corpus size may explain this, most severely impacting shorter-sequence models. Confirming this hypothesis, Instructor-XL and E5-Large-V2 showed similar diversity given identical sequence lengths (Table 1).

Fig. 1 shows the topic coherence and diversity across embedding models by mapping them along these two dimensions for each topic modeling pipeline. This representation reveals clear distinctions in the attainable average outcome space per model. The two-dimensional mapping elucidates underlying trade-offs between competing performance metrics when evaluating neural topic modeling capabilities across embedding approaches. Observation indicates no definitive performance advantage stemming solely from greater maximum input sequence lengths and their associated higher embedding model parameterizations. De-

⁴ <https://github.com/MaartenGr/BERTopic>

⁵ <https://github.com/MIND-Lab/OCTIS/tree/master>

spite proportionately sized representations, interference from additional pipeline components likely confounds any potential gains at the embedding stage.

Table 3: Average performance due to Using different Embedding Models

Embedding Model	NPMI	TD
Instructor-XL	0.0396	0.6844
SGPT-5.8B	0.0324	0.7234
E5-Large-V2	0.0283	0.6890
All-MiniLM-L6-V2	0.0122	0.7223

Table 4 shows averaging across other pipeline components indicates forgoing dimension reduction (No DR) maximizes coherence (0.0392) while barely affecting diversity. UMAP achieved highest topic diversity (0.7619) despite using the same target dimensions ($n=5$) as Truncated-SVD. However, omitting dimensionality reduction substantially prolongs topic modeling training time, with observed runtimes exhibiting an approximately 2-fold increase relative to pipelines incorporating dimensionality reduction.

Table 4: Effect of DR on average performance

DR Method	NPMI	TD	Time(s)
No DR	0.0392	0.7263	416.9
Truncated-SVD	0.0257	0.6261	286.5
UMAP	0.0194	0.7619	298.8

Table 5 shows K-Means achieved superior coherence (0.0445) while HDBSCAN improved topic diversity (0.7583). HDBSCAN’s noise-sensitivity enables it to be outlier-aware compared to K-Means. This allows HDBSCAN to potentially discover a greater number of distinct topics within complex textual data containing outliers. Conversely, K-Means clusters may densify topics by retaining outliers, improving coherence but reducing unique content. These observations suggest cluster cohesion influences topic modeling performance differences between hard and soft clustering approaches.

Table 5: Effect of clustering models on NPMI and TD on average

Clustering Algorithm	NPMI	TD
K-Means	0.0445	0.6513
HDBSCAN	0.0117	0.7583

Results indicate the optimal combination of model components to maximize topic coherence is *normalized corpus*, Instructor-XL embedding model, no dimensionality reduction, and K-means clustering (0.1085 NPMI, 0.6357 TD). This configuration achieved the highest coherence score (0.1085) while also attaining relatively high topic diversity (0.6357) as shown in Table 6. In contrast, maximizing topic diversity specifically requires fitting the *normalized corpus* with the

SGPT-5.8B embedding model, UMAP for dimensionality reduction, and HDBSCAN clustering based on the highest attained topic diversity score (0.9136). Worst coherence (-0.0631) occurred with the *improved corpus*, All-MiniLM-L6-V2, Truncated SVD, and HDBSCAN, though this layout still surpassed top NPMI model in diversity (0.7416 TD). Most topic overlap (0.4643 TD) resulted from the *improved corpus* with Instructor-XL, Truncated SVD, and K-Means.

5 Conclusions

This study has highlighted current gaps in optimizing neural topic modeling pipelines, specifically the selection of embedding models in conjunction with dimensionality reduction and clustering components, for complex real-world data. The experimental results indicate text normalization improves both coherence and diversity metrics. Specifically, the context-aware Instructor-XL embeddings maximize topic coherence, outperforming larger models like SGPT-5.8B which generate more diverse topics. This aligns with findings that maintaining higher embedding dimensions boosts coherence at the expense of substantially prolonged model fitting. Regarding clustering methods, the noise-sensitive HDBSCAN algorithm enhances topic diversity by excluding outlier documents to be included in topics, while reducing coherence, with the inverse pattern observed for the strictly centroid-based K-Means approach which includes outliers. Thus, optimization of the underlying neural topic modeling pipeline components enables balancing performance trade-offs. Through comprehensive evaluation, we established best practice guidelines for configuring neural topic modeling pipelines. Specifically, the optimal configuration for maximizing topic coherence consisted of *normalized corpus*, the Instructor-XL for generating embeddings, no dimensionality reduction, and K-Means clustering. This combination yielded the highest coherence scores, when applied to our complex real-world dataset. In contrast, the maximum attainable topic diversity resulted from a different pipeline layout of *normalized corpus*, SGPT-5.8B embeddings, UMAP for dimensionality reduction, and HDBSCAN clustering. Notably, this demonstrates that separate component configurations are required to independently optimize either coherence or diversity, highlighting an inherent trade-off between the two desired outcomes for neural topic models.

This research allowed us to identify a number of open issues, which we plan to address in subsequent work. We offered insights for topic modeling of large and noisy transcripts, however findings may be constrained by automatic speech recognition deficiencies. Moreover, real-world data features topics peripheral to telephony conversations’ main themes. Distinguishing speaker roles or converting dialogues into third-person narratives could illuminate effects on topic quality and reducing outlier documents. Future work should explore tailored prompt engineering and parameter tuning for generating specialized embeddings, as well as mitigation strategies for overcoming speech recognition errors and dialog digressions that may constrain topic discovery. Additionally, further research will focus on fine-tuning automated speech recognition systems to specialize in accu-

Table 6: NPMI and Topic Diversity score for different components

	Corpus	Embedding Model	DR	Clustering	NPMI	TD
1	Normalized corpus	Instructor-Xl	No DR	K-Means	0.1085	0.6357
2	Normalized corpus	Instructor-Xl	Truncated-SVD	K-Means	0.1050	0.4940
3	Normalized corpus	E5-Large-V2	No DR	K-Means	0.0796	0.7497
4	Normalized corpus	All-MiniLM-L6-V2	No DR	K-Means	0.0776	0.7105
5	Normalized corpus	SGPT-5.8B	No DR	K-Means	0.0745	0.7409
6	Normalized corpus	All-MiniLM-L6-V2	UMAP	K-Means	0.0709	0.6988
7	Normalized corpus	Instructor-Xl	UMAP	K-Means	0.0708	0.7172
8	Improved Corpus	SGPT-5.8B	Truncated-SVD	HDBSCAN	0.0693	0.6172
9	Normalized corpus	E5-Large-V2	UMAP	K-Means	0.0622	0.7270
10	Improved Corpus	Instructor-Xl	Truncated-SVD	HDBSCAN	0.0589	0.6078
11	Normalized corpus	SGPT-5.8B	UMAP	K-Means	0.0566	0.7202
12	Normalized corpus	SGPT-5.8B	Truncated-SVD	K-Means	0.0566	0.5836
13	Normalized corpus	All-MiniLM-L6-V2	No DR	HDBSCAN	0.0556	0.7871
14	Normalized corpus	E5-Large-V2	Truncated-SVD	K-Means	0.0509	0.5617
15	Normalized corpus	Instructor-Xl	Truncated-SVD	HDBSCAN	0.0434	0.7991
16	Normalized corpus	All-MiniLM-L6-V2	Truncated-SVD	K-Means	0.0403	0.5178
17	Improved Corpus	All-MiniLM-L6-V2	No DR	K-Means	0.0390	0.7341
18	Improved Corpus	Instructor-Xl	No DR	K-Means	0.0385	0.6193
19	Normalized corpus	E5-Large-V2	No DR	HDBSCAN	0.0379	0.7664
20	Improved Corpus	E5-Large-V2	No DR	K-Means	0.0305	0.7518
21	Improved Corpus	SGPT-5.8B	No DR	HDBSCAN	0.0296	0.6508
22	Normalized corpus	Instructor-Xl	No DR	HDBSCAN	0.0271	0.8055
23	Normalized corpus	Instructor-Xl	UMAP	HDBSCAN	0.0266	0.8982
24	Normalized corpus	SGPT-5.8B	No DR	HDBSCAN	0.0264	0.8512
25	Improved Corpus	SGPT-5.8B	No DR	K-Means	0.0259	0.7351
26	Improved Corpus	E5-Large-V2	UMAP	HDBSCAN	0.0240	0.6563
27	Normalized corpus	SGPT-5.8B	UMAP	HDBSCAN	0.0207	0.9136
28	Normalized corpus	E5-Large-V2	Truncated-SVD	HDBSCAN	0.0194	0.7358
29	Normalized corpus	E5-Large-V2	UMAP	HDBSCAN	0.0194	0.8639
30	Improved Corpus	SGPT-5.8B	Truncated-SVD	K-Means	0.0188	0.5646
31	Improved Corpus	All-MiniLM-L6-V2	Truncated-SVD	K-Means	0.0154	0.5308
32	Improved Corpus	Instructor-Xl	Truncated-SVD	K-Means	0.0153	0.4643
33	Improved Corpus	E5-Large-V2	Truncated-SVD	HDBSCAN	0.0138	0.5377
34	Improved Corpus	Instructor-Xl	UMAP	K-Means	0.0100	0.7236
35	Improved Corpus	All-MiniLM-L6-V2	UMAP	K-Means	0.0069	0.6899
36	Improved Corpus	SGPT-5.8B	UMAP	K-Means	0.0064	0.6969
37	Normalized corpus	SGPT-5.8B	Truncated-SVD	HDBSCAN	0.0058	0.8607
38	Improved Corpus	E5-Large-V2	UMAP	K-Means	0.0046	0.7150
39	Improved Corpus	E5-Large-V2	Truncated-SVD	K-Means	0.0032	0.5492
40	Improved Corpus	SGPT-5.8B	UMAP	HDBSCAN	-0.0017	0.7463
41	Improved Corpus	All-MiniLM-L6-V2	No DR	HDBSCAN	-0.0056	0.7400
42	Improved Corpus	E5-Large-V2	No DR	HDBSCAN	-0.0060	0.6529
43	Improved Corpus	Instructor-Xl	No DR	HDBSCAN	-0.0112	0.6904
44	Normalized corpus	All-MiniLM-L6-V2	UMAP	HDBSCAN	-0.0153	0.8815
45	Improved Corpus	Instructor-Xl	UMAP	HDBSCAN	-0.0172	0.7581
46	Improved Corpus	All-MiniLM-L6-V2	UMAP	HDBSCAN	-0.0337	0.7840
47	Normalized corpus	All-MiniLM-L6-V2	Truncated-SVD	HDBSCAN	-0.0422	0.8518
48	Improved Corpus	All-MiniLM-L6-V2	Truncated-SVD	HDBSCAN	-0.0631	0.7416

rate transcription of real-world telephony conversations. Enhancing recognition performance could improve topic modeling capabilities. Overcoming these limitations may substantially improve discovery and interpretation of topics within complex real-world textual data.

Acknowledgments The authors would like to thank Merseyside Police UK for providing the data and for assistance in deeply comprehending the corpus.

Disclosure of Interests The authors declare no relevant competing interests related to the content presented in this article.

References

1. Arazzi, M., Ferretti, M., Nocera, A.: Analysis of Video Lessons: A Case for Smart Indexing and Topic Extraction. In: CEUR WORKSHOP PROCEEDINGS. vol. 3340, pp. 156–167 (2022)
2. Blei, D.M.: Probabilistic topic models. *Communications of the ACM* **55**(4), 77–84 (2012). <https://doi.org/10.1145/2133806.2133826>
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *J. Mach. Learn. Res* **3**, 993–1022 (2003)
4. Bora, D.J., Gupta, D.A.K.: A comparative study between fuzzy clustering algorithm and hard clustering algorithm. *arXiv preprint arXiv:1404.6059* (2014). <https://doi.org/10.14445/22312803/IJCTT-V10P119>
5. Dieng, A.B., Ruiz, F.J., Blei, D.M.: Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics* **8**, 439–453 (2020)
6. Errattahi, R., El Hannani, A., Ouahmane, H.: Automatic speech recognition errors detection and correction: A review. *Procedia Computer Science* **128**, 32–37 (2018). <https://doi.org/10.1016/j.procs.2018.03.005>
7. Falini, A.: A review on the selection criteria for the truncated SVD in Data Science applications. *Journal of Computational Mathematics and Data Science* p. 100064 (2022). <https://doi.org/10.1016/j.jcmds.2022.100064>
8. Grootendorst, M.: BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv preprint arXiv:2203.05794* (2022). <https://doi.org/10.48550/arXiv.2203.05794>
9. Khalid, H., Wade, V.P.: Comparative analysis of existing and a novel approach to topic detection on conversational dialogue data. *International Journal on Natural Language Computing* (2022)
10. Kherwa, P., Bansal, P.: Topic modeling: A comprehensive review. *EAI Endorsed Trans. Scalable Inf. Syst.* **7**(24) (2019). <https://doi.org/10.4108/eai.13-7-2018.159623>
11. Köster, F.: *Multidimensional Analysis of Conversational Telephone Speech*. Springer (2018)
12. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. vol. 1, pp. 281–297. Oakland, CA, USA (1967)
13. McInnes, L., Healy, J., Astels, S.: Hdbscan: Hierarchical density based clustering. *J. Open Source Softw.* **2**(11), 205 (2017). <https://doi.org/10.21105/joss.00205>

14. McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018). <https://doi.org/10.48550/arXiv.1802.03426>
15. Muennighoff, N.: Sgpt: Gpt sentence embeddings for semantic search. arXiv preprint arXiv:2202.08904 (2022). <https://doi.org/10.48550/arXiv.2202.08904>
16. Muennighoff, N., Tazi, N., Magne, L., Reimers, N.: MTEB: Massive text embedding benchmark. arXiv preprint arXiv:2210.07316 (2022). <https://doi.org/10.48550/arXiv.2210.07316>
17. Nanga, S., Bawah, A.T., Acquaye, B.A., Billa, M.I., Baeta, F.D., Odai, N.A., Obeng, S.K., Nsiah, A.D.: Review of dimension reduction methods. *Journal of Data Analysis and Information Processing* **9**(3), 189–231 (2021). <https://doi.org/10.4236/jdaip.2021.93013>
18. Pan, D., Yang, Z., Tan, H., Wu, J., Lin, H.: Dialogue topic extraction as sentence sequence labeling. In: *CCF International Conference on Natural Language Processing and Chinese Computing*. pp. 252–262. Springer (2022)
19. Papadia, G., Pacella, M., Perrone, M., Giliberti, V.: A Comparison of Different Topic Modeling Methods through a Real Case Study of Italian Customer Care. *Algorithms* **16**(2), 94 (2023). <https://doi.org/10.3390/a16020094>
20. Radford, A., Kim, J.W., Xu, T., Brockman, G., McLeavey, C., Sutskever, I.: Robust speech recognition via large-scale weak supervision. In: *International Conference on Machine Learning*. pp. 28492–28518. PMLR (2023)
21. Röder, M., Both, A., Hinneburg, A.: Exploring the space of topic coherence measures. In: *Proceedings of the 8th ACM International Conference on Web Search and Data Mining*. pp. 399–408 (2015). <https://doi.org/10.1145/2684822.2685324>
22. Su, H., Kasai, J., Wang, Y., Hu, Y., Ostendorf, M., Yih, W.t., Smith, N.A., Zettlemoyer, L., Yu, T.: One embedder, any task: Instruction-finetuned text embeddings. arXiv preprint arXiv:2212.09741 (2022). <https://doi.org/10.48550/arXiv.2212.09741>
23. Tshimula, J.M., Nkashama, D.K., Owusu, P., Frappier, M., Tardif, P.M., Kabanza, F., Brun, A., Patenaude, J.M., Wang, S., Chikhaoui, B.: Characterizing Financial Market Coverage using Artificial Intelligence. arXiv preprint arXiv:2302.03694 (2023). <https://doi.org/10.48550/arXiv.2302.03694>
24. Vasconcellos, P.H.S., Lara, P.D.d.A., Marques-Neto, H.T.: Analyzing Polarization And Toxicity On Political Debate In Brazilian TikTok Videos Transcriptions. In: *Proceedings of the 15th ACM Web Science Conference*. pp. 33–42 (2023). <https://doi.org/10.1145/3578503.3583613>
25. Vayansky, I., Kumar, S.A.: A review of topic modeling methods. *Information Systems* **94**, 101582 (2020). <https://doi.org/10.1016/j.is.2020.101582>
26. Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., Wei, F.: Text embeddings by weakly-supervised contrastive pre-training. arXiv preprint arXiv:2212.03533 (2022). <https://doi.org/10.48550/arXiv.2212.03533>
27. Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., Zhou, M.: Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems* **33**, 5776–5788 (2020)
28. Zoupanos, S., Kolovos, S., Kanavos, A., Papadimitriou, O., Maragoudakis, M.: Efficient comparison of sentence embeddings. In: *12th Hellenic Conference on Artificial Intelligence*. pp. 1–6 (2022). <https://doi.org/10.1145/3549737.3549752>