KeyBERT for Dynamic Topic Modeling and Summarization

Davide Elio Stefano Demicheli, Annalisa Belloni, Valeria Petrianni, Valerio Xefteris

***Abstract*—Keyword extraction plays a crucial role in various natural language processing (NLP) tasks, providing a compact representation of textual content. In this work, we explore two novel extensions of KeyBERT [1] to enhance dynamic topic modeling and abstractive text summarization. The first extension introduces a dynamic topic modeling pipeline designed to track topic evolution over time in news articles. By extracting keywords from news headlines and clustering them, our approach identifies major topics and their temporal dynamics efficiently. Results on Australian ABC News data demonstrate the pipeline's ability to capture the temporal evolution of events such as the COVID-19 pandemic, major environmental disasters, political shifts, supported by word clouds to enhance topic interpretability. Despite challenges related to stopword handling and memory constraints, the method offers a practical trade-off between efficiency and quality of results. The second extension investigates the impact of keyword-enriched text representations on abstractive summarization. By prepending extracted keywords to the input text and fine-tuning a summarization model, we evaluate two annotation strategies: in the *LIST* approach, all keywords are grouped together at the beginning of the text, enclosed within special tokens; in the *SINGLE* approach, each keyword is instead individually highlighted within special tokens. Empirical results show that both strategies yield small but consistent improvements over the baseline, particularly in recall and F1-score. Manual qualitative analysis further supports the effectiveness of keyword integration in improving the quality of the summary. Overall, our study highlights the potential of leveraging keyword extraction to enhance topic modeling and summarization, also paving the way for more interpretable NLP applications. All the code developed for this project, along with the instructions for running it, is available at: https://github.com/demidavi7/KeyBERT_Topic-Modelling_Summarization.**

I. KeyBERT

KeyBERT [1] is a keyword extraction tool that leverages transformer-based embeddings, particularly BERT, to identify the most relevant words or phrases in a given text. It works by generating a document embedding and comparing it to n-gram embeddings, ranking them based on cosine similarity. This allows KeyBERT to efficiently extract meaningful keywords that best represent the content of a text, making it useful for tasks such as topic modeling, summarization, and information retrieval.

II. First extension

A. Problem statement

The first extension of our study employs KeyBERT to analyze the topics identified within a dataset containing news headlines published over a period of several years. Our objective is to identify the key topics covered throughout this period and examine their evolution over time, where each topic is defined by a set of keywords that share a high degree of semantic similarity.

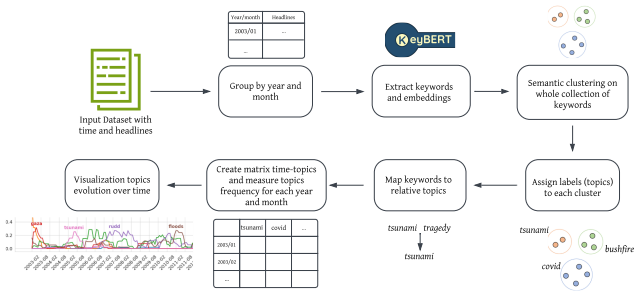

Fig. 1. Pipeline for Dynamic Topic Modeling

B. Methodology

In Fig. 1, we present a high-level overview of the pipeline illustrating the steps followed in the implementation of the first extension.

As a first step, we defined a set of stopwords tailored to the specific requirements of our task. Given that the dataset we utilized originates from an Australian news source, we sought to prevent the extracted topics from being overly centered on individual cities. While geographic references are relevant in many contexts, our primary focus was on broader events, such as environmental disasters, political developments, and other significant occurrences. To achieve this, we removed the names of major cities in Oceania, along with other terms deemed irrelevant for our analysis. In addition, we filtered out standard stopwords to ensure that only meaningful terms contributed to topic extraction.

After preprocessing, we grouped the headlines into monthly collections based on their respective month and year. We then applied KeyBERT separately to each group to extract keywords while retaining their embeddings. The embeddings corresponding to these keywords were subsequently clustered using HDBSCAN, allowing us to form groups of semantically related keywords that would serve as the defining terms for each topic.

To determine the optimal hyper-parameter configuration for the clustering, we conducted a grid search using a custom evaluation metric. This metric was defined as a convex combination of two factors: the average silhouette score $\bar{s}$, computed on the cosine similarity matrix [2], and the average probability $\bar{p}$ of data points to belong to the assigned cluster.

$$\text{score} = \alpha\bar{s} + (1-\alpha)\bar{p} \tag{1}$$

Once the optimal hyper-parameters were identified, we evaluated the resulting clusters using the same metric.

Each month-year group was then assigned a list of topics by matching the keywords in its headlines with those defining each cluster. To assign a representative keyword as the label for a topic cluster, we employed a weighted scoring approach that combines centrality and frequency. Given a set of keywords belonging to a cluster, we computed a weighted centroid, where each keyword's embedding is weighted by its frequency. The centrality of each keyword is measured using cosine similarity to this centroid. The final score is computed as a weighted sum of centrality and normalized frequency, with tunable weights. The keyword with the highest score is selected as the representative topic label. Finally, to analyze trends, we first computed a time-topic matrix collecting the normalized frequencies of the topics over time, then we visualized the temporal dynamics of the most significant topics, focusing on the top *spiking* topics, which are topics that exhibited the most pronounced peaks in popularity at specific moments in time.

C. Experiments

We utilized the A Million News Headlines [3] Dataset, which contains news headlines published over a span of nineteen years. This dataset is sourced from the ABC (Australian Broadcasting Corporation), a reputable Australian news outlet, and is provided in CSV format as a single file.

The dataset includes the following fields:

- ***publish_date***: The date the article was published, formatted as yyyyMMdd.
- ***headline_text***: The text of the headline, which is in ASCII, English, and lowercase.

The time range of the dataset extends from February 19, 2003 to December 31, 2021, encompassing a total of 1,213,004 samples. For the purposes of running our pipeline, we extracted a sample of 1,000 articles for each year-month period.

KeyBERT is used to extract 50 keywords for each year-month period, capturing both unigrams and bigrams to identify key themes over time.

The hyper-parameters we tuned for HDBSCAN are:

- ***min_cluster_size***: the minimum number of samples in a group for that group to be considered a cluster; groupings smaller than this size will be left as noise.
- ***min_samples***: the minimum number of neighbours to a core point. The higher this is, the more points are going to be discarded as noise/outliers.

The tested values are detailed in Table I. Note that only combinations where *min_cluster_size* was greater or equal to *min_samples* were considered. The best configuration, using as parameter for the evaluation metric in (1) $\alpha = 0.45$, is given by *min_cluster_size* = 20 and *min_samples* = 10.

Fig. 2 illustrates the temporal evolution of the seven most prominent spiking topics identified by our pipeline. The most dominant topic is *coronavirus covid*, which exhibits a sharp increase in frequency starting in early 2020, aligning precisely with the onset of the COVID-19 pandemic. Another notable topic is *rudd*, which sees a peak during the years corresponding to Kevin Rudd's tenure as Prime Minister of Australia (2007-2010). This indicates that the model correctly identifies significant political figures when they are most relevant in public discourse. Additionally, the topic *bushfire* experiences pronounced spikes that coincide with Australia's major bushfire crises, notably the 2009 Black Saturday fires and the devastating 2019-2020 bushfire season. This alignment further demonstrates the model's ability to track environmental disasters over time. The presence of *Gaza* as a topic suggests that international conflicts like the Second Intifada also gained significant media attention during specific periods. To further investigate topic coherence and interpretability, we also analyzed word clouds for specific topics (see Appendix A). Overall, these results confirm that our dynamic topic modeling pipeline effectively captures major events, political shifts, and environmental crises, highlighting its potential for tracking news trends over time.

TABLE I
TESTED VALUES FOR HYPERPARAMETERS IN HDBSCAN

Parameter	Values
min_cluster_size	{10, 15, 20, 25, 30}
min_samples	{10, 15, 20, 25}

D. Conclusions

In this work, we proposed a dynamic topic modeling approach based on keyword extraction with KeyBERT and semantic clustering. The method demonstrated several advantages. First, it is inherently incremental: the most computationally expensive step, keyword extraction, can be performed only on newly added documents, making it well-suited for real-time applications. While KeyBERT and HDBSCAN are not inherently the most efficient methods, the pipeline remains practical when applied to a limited dataset and a controlled number of extracted keywords. This constraint, which is necessary to avoid memory issues, allows the process to run in a reasonable time—approximately 10 minutes on a GPU—making it feasible even with moderate computational resources. The qualitative analysis showed that the method produces reasonable and meaningful results, successfully capturing major events and trends over time. Furthermore, it offers a degree of customizability, as modifying the stopword list allows the model to focus on specific trends of interest. Finally, word clouds enhance interpretability by providing insight into the composition of topic clusters. Despite these strengths, the method also presents some challenges. The handling of stopwords remains a partially manual process, requiring human intervention to refine and adjust the list of excluded terms. While large language models (LLMs) could assist in this task, human oversight is still necessary to ensure relevance. Another limitation is the memory-intensive nature of the approach: storing high-dimensional dense embeddings for all extracted keywords imposes constraints on the dataset size and the number of keywords that can be processed. Additionally, the

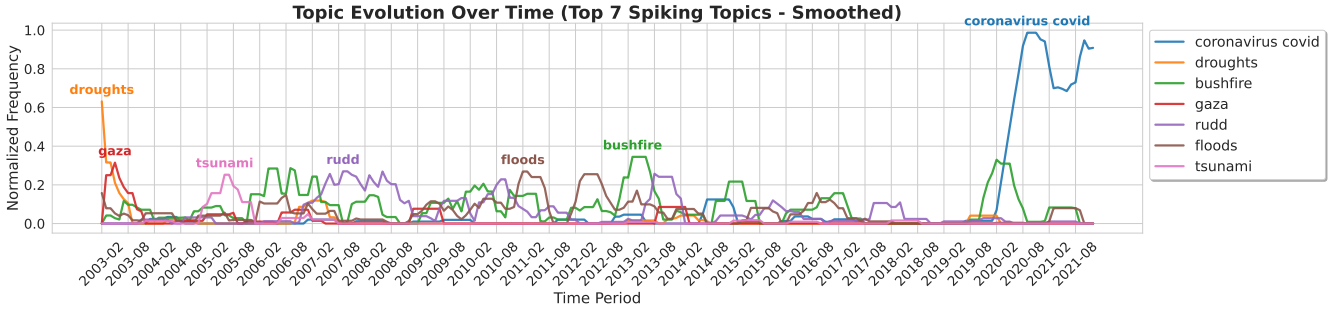

Fig. 2. Evolution over time of the top 7 spiking topics

method relies on the quality of both the extracted keywords and the clustering process, meaning that inaccuracies in either step could affect the final topic assignments. Overall, our pipeline achieves a reasonable balance between quality of results, efficiency, interpretability, and adaptability, making it a viable solution for tracking evolving trends in news data. Future work could explore more efficient embedding storage techniques, automated stopword refinement, and alternative clustering strategies to further enhance scalability and robustness.

III. Second Extension

A. Problem Statement

The primary goal of this second extension is to improve abstractive text summarization leveraging keywords generated with KeyBERT. The idea is to enrich the input text by concatenating the original text with a series of highlighted keywords extracted from it, to help the summarization model better capture the key concepts.

B. Methodology

Our pipeline comprises three main stages: data preparation with keyword extraction, keyword-enriched models fine-tuning, and final inference for summary generation.

To perform abstractive summarization, we leveraged the `sshleifer/distilbart-xsum-6-6` model [4], which is a fine-tuned version of DistilBART trained on the XSum corpus (an English-language summarization dataset covering a broad range of news topics). For our experiments, which included an additional fine-tuning of `sshleifer/distilbart-xsum-6-6`, we chose to work with the CNN/DailyMail dataset [5] (a summarization dataset also provided in English and related to the news domain, but not overlapping with XSum) in order to prevent the risk of overfitting during the fine-tuning phase.

As a first step, we sampled a subset of 50,000 records from the CNN/DailyMail dataset. This subset, of approximately 200 MB in size, provided a sufficiently large yet manageable corpus for experimentation. For each record, after extracting the article and its reference summary, we utilized the KeyBERT model to extract the most relevant terms from each article.

To strike a balance between brevity and specificity, we set the parameter `keyphrase_ngram_range=(1, 2)`, allowing the extraction of both single-word and two-word keywords. Regarding the number of keywords to extract, we selected `top_n=10` to ensure comprehensive topic coverage while avoiding noise. Extracting too many keywords could introduce unnecessary redundancy, whereas extracting too few might omit critical concepts. To further refine keyword selection, we enabled the Maximal Marginal Relevance method by setting `use_mmr=True` and configured `diversity=0.5`, reducing redundancy among extracted keywords while maintaining moderate diversity to preserve the core semantic meaning of the text. Finally, we filtered out common English stop words. Adopting a consistent tagging scheme, these keywords became "special tokens" that a downstream summarization model could then learn to recognize explicitly. Once the relevant keywords were extracted, we created two variants of the enriched dataset. These were then used to fine-tune the chosen summarization model, allowing us to investigate different keywords integrations. In both variants the keywords were placed at the beginning of the article text to ensure the model could immediately attend to them. These are:

- *Top Prepending*: each extracted keyword is wrapped in a `<keyword>...</keyword>` tag, allowing the model to learn to recognize the highlighted semantic information with precise granularity;
- *Top Prepending with List*: the entire keyword list is enclosed within a single `<keyword>...</keyword>` tag, training the model to interpret the entire block of keywords as a unified representation of the highlighted semantic information.

The two resulting datasets, where each record consists of a keywords-enriched article text and its reference summary, were published as Hugging Face datasets [6], [7].

We conducted experiments with three distinct setups, all originating from the same DistilBART checkpoint provided by the `sshleifer/distilbart-xsum-6-6` model:

- the baseline model, hereafter referred to as BASE, was obtained by fine-tuning the chosen model on the simple CNN/DailyMail texts, without any additional keyword annotations;
- the second model, hereafter referred to as SINGLE, was built by fine-tuning the chosen model on the *Top Prepending* version of the enriched dataset;

- finally, the last model, hereafter referred to as LIST, was obtained by fine-tuning the chosen model on the *Top Prepending with List* version of the enriched dataset.

The decision to fine-tune the base model on the raw CNN/DailyMail dataset was made to allow for a meaningful assessment of our pipeline's effectiveness. This approach ensures in fact that any improvements observed in the base model can be directly attributed to the use of enriched text, rather than being influenced by domain adaptation factors.

We conducted a lightweight fine-tuning across all models, ensuring that the specifications were consistent for all three, so that the results could be compared meaningfully, without being influenced by variations in the fine-tuning process.

Specifically, we froze most of the parameters in the inherited `sshleifer/distilbart-xsum-6-6` model, only allowing the token embedding layer, the language modeling head, and the final two decoder layers to be updated.

Furthermore, we modified the loss function originally used in the `sshleifer/distilbart-xsum-6-6` model by introducing a multi-component loss function to guide our models' improvement. This new loss function combines cross-entropy loss to maintain alignment with the reference summaries at the token level, with additional terms based on ROUGE and BERTScore, which contribute to enhancing both syntactic and semantic coverage. We defined the total loss $\mathcal{L}_{\text{combined}}$ as a weighted sum of three components:

$$\mathcal{L}_{\text{combined}} = \alpha \cdot \mathcal{L}_{\text{CE}} + \beta \cdot \mathcal{L}_{\text{ROUGE}} + \gamma \cdot \mathcal{L}_{\text{BERT}}, \qquad (2)$$

where the cross-entropy loss $\mathcal{L}_{\text{CE}}$ enforces token-level accuracy, while $\mathcal{L}_{\text{ROUGE}} = 1 - \overline{\text{ROUGE\_F1}}$ and $\mathcal{L}_{\text{BERT}} = 1 - \text{BERTScore\_F1}$ encourage respectively syntactic alignment and thematic coverage. In particular, $\overline{\text{ROUGE\_F1}}$ is given by the mean over the three different F1-scores associated to the three considered ROUGE metrics (ROUGE-1, ROUGE-2 and ROUGE-L). Finally, the coefficients were set to $\alpha = 0.4$, $\beta = 0.3$, and $\gamma = 0.3$ to balance lexical precision with high-level coherence. Moreover, only for the two models distinct from the baseline working with keywords-enriched texts, we extended the tokenizer's vocabulary to incorporate the newly introduced `<keyword>` and `</keyword>` tokens, ensuring that the model could learn specialized embeddings for these elements.

C. Experiments

We now provide a brief summary of the main hyperparameters involved in the fine-tuning phase, together with some technical details:

- **Training and Evaluation Data:** we performed a split of 85% training and 15% evaluation of the sampled data.
- **Number of Epochs:** we used num_train_epochs $= 2$, since this provided enough time for the network to learn from the keyword-augmented data without risking overfitting.
- **Optimizer:** we employed the Adafactor optimizers, for its memory efficiency quality.
- **Batch Size:** the batch size was set to 16.
- **Learning Rate:** we set learning_rate $= 1 \times 10^{-5}$, which is slightly lower than the one from the original model (which was equal to 3×10^{-5}), in order to avoid the risk of destabilizing too much its pre-trained weights.
- **Weight Decay:** we set the regularization term weight_decay $= 0.01$, in order to mitigate overfitting by penalizing large parameter weights.
- **GPU and Training Time:** the fine-tuning process was carried out on 2 x T4 GPUs in Kaggle, and each model required approximately 1 hour and 40 minutes.

To evaluate the syntactic quality of the summaries, we used ROUGE metrics, while for assessing semantic alignment, we employed BERTScore. The results are reported in Table II.

Compared to the baseline, both keyword-enriched variants, SINGLE and LIST, show overall slight improvements, particularly in recall and F1-score.

TABLE II
EVALUATION ON 1,000 TEST DOCUMENTS FROM CNN/DAILYMAIL

Metric	BASE	SINGLE	LIST
ROUGE-1 Precision	0.4603	**0.4672**	0.4603
ROUGE-1 Recall	0.3053	0.3235	**0.3326**
ROUGE-1 F1-score	0.3508	0.3656	**0.3689**
ROUGE-2 Precision	0.1876	0.1962	**0.1970**
ROUGE-2 Recall	0.1206	0.1320	**0.1370**
ROUGE-2 F1-score	0.1401	0.1509	**0.1542**
ROUGE-L Precision	0.3161	**0.3161**	0.3135
ROUGE-L Recall	0.2071	0.2171	**0.2246**
ROUGE-L F1-score	0.2389	0.2459	**0.2498**
BERTScore Precision	0.8894	**0.8896**	0.8893
BERTScore Recall	0.8663	0.8690	**0.8701**
BERTScore F1-score	0.8775	0.8790	**0.8794**

The LIST model consistently achieves the highest recall and F1-score across ROUGE-1, ROUGE-2, and ROUGE-L, suggesting that consolidating all keywords into a single block helps the model capture more relevant information. SINGLE model achieves slightly higher precision in some metrics, but it does not surpass LIST in recall or overall F1-score. BERTScore remains relatively consistent across all configurations. A qualitative analysis of the results is conducted in Appendix B.

D. Conclusions

In this work, we explored different ways to leverage keywords generated by KeyBERT to finetune a summarization model to recognize and utilize them, experimenting with two distinct integration formats. Our results indicate that enriching the input text with extracted keywords provides a strong cue for improving summarization quality. One of the main challenges is the computational cost of KeyBERT, particularly evident when processing long texts. Future research could explore alternative fine-tuning strategies, such as varying the number of trainable layers, utilizing a different optimizer and learning rate, or training for higher number of epochs. Another interesting cue could be testing alternative methods for emphasizing keywords. For instance, instead of duplicating them at the beginning of the text, they could be highlighted within the text itself.

APPENDIX A
FIRST EXTENSION

Fig. 3 contains an example of word cloud for the *tsunami* cluster, highlighting the most frequent words. This kind of visualization provides additional insights into the topic's context, which is otherwise only represented by a single term in the temporal evolution graph. Moreover, the word cloud serves as a means to visually inspect cluster coherence, ensuring that the extracted topics meaningfully group related terms.

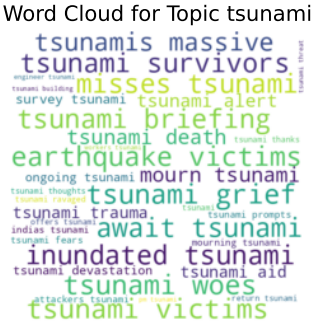

Fig. 3. Example of word cloud for the cluster 'tsunami'

APPENDIX B
SECOND EXTENSION

Even if SINGLE and LIST models achieve timid improvements compared to our baseline regarding syntactic and semantic metrics, to concretely illustrate the impact of our pipeline on the output quality, we reported a test example that well demonstrates the differences among the models. As shown in Table III, the BASE model emphasizes the club's hope to see restrictions lifted soon and compares City's spending to that of other prominent teams, yet it does not capture the role of UEFA in punishing the club for breaching Financial Fair Play rules. Both the SINGLE and the LIST models, instead, thanks to the support given by the extracted keywords ("UEFA", "FINANCIALLY", "TRANSFER PENALTIES", "FPP REGULATIONS") are able to recognize the importance of UEFA's punishment to Manchester City in the summary they generate. At the same time, due to the presence of keywords like "LIVERPOOL CONTRACT", "ALLOW STERLING", the two models both refer to major player signings, even if these considerations are not present in the ground truth. This result is interesting because it might be interpreted as a sign that the model has effectively learned to give more importance to the keywords, but it is also true that this behavior might become a problem if the keywords are characterized by poor quality.

TABLE III
SAMPLE SUMMARIES GENERATED BY THREE DIFFERENT MODEL CONFIGURATIONS AND THE REFERENCE TEXT

Model	Summary
REFERENCE	"Manchester City have been restricted to a net transfer spend of £49m. The club also had to keep its overall wage bill to the current level of £205m. These punishments were imposed by UEFA for breaching Financial Fair Play rules. The spending restrictions apply to this season and the next one, but City are confident they will be lifted early after their compliance."
BASE	"Manchester City are hoping the club's financial restrictions will be lifted at the end of the season. Pellegrini says they will be in the same position as the other English clubs have been. Manchester United, Barcelona, Liverpool and Arsenal have all paid more in transfer fees than City in the past 12 months."
SINGLE	"Manchester City are confident they will be able to bid for stars like Raheem Sterling and Gareth Bale this summer. Manchester City have been hampered over the past year by financial restrictions imposed by Uefa. The governing body says they have been financially compliant with FFP rules."
LIST	"Manchester City are confident they will be able to bid for stars like Raheem Sterling, Gareth Bale and Ross Barkley this summer. Pellegrini is confident the club's transfer restrictions will be lifted at the end of the season. Premier League champions have been hampered over the past year by Uefa's financial restrictions on spending and wages."

REFERENCES

[1] M. Grootendorst, "Keybert: Minimal keyword extraction with bert." https://github.com/MaartenGr/KeyBERT, 2020.
[2] P. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. comput. appl. math. 20, 53-65," *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, 11 1987.
[3] "A million news headlines," https://www.kaggle.com/datasets/therohk/million-headlines.
[4] https://huggingface.co/sshleifer/distilbart-xsum-6-6, 2021.
[5] https://www.kaggle.com/datasets/gowrishankarp/newspaper-text-summarization-cnn-dailymail, 2022.
[6] https://huggingface.co/datasets/VexPoli/cnn_enrich_with_top_keywords?row=5, 2025.
[7] https://huggingface.co/datasets/VexPoli/cnn_enrich_with_top_keywords_modified?row=0, 2025.