LDK 2023

Domain-Specific Keyword Extraction using BERT

Jill Sammet
Kiel University
Kiel, Germany
jillsammet@web.de

Ralf Krestel
ZBW - Leibniz Information Centre
for Economics & Kiel University
Kiel, Germany
rkr@informatik.uni-kiel.de

Abstract

Maintaining domain-specific thesauri is a costly endeavor. Terms might get added, removed, or merged over time to reflect new trends and keep the thesaurus consistent. This work is done by domain experts following predefined rules. Instead of curating the thesaurus manually, we investigate the use of language models to automatically propose novel terms to be added. To this end, we present an approach for keyword extraction from titles and abstracts of domain-specific documents. We report results on fine-tuned BERT models and compare them with different baselines. We further show that our proposed approach outperforms others in various evaluation scenarios.

1 Introduction

The Thesaurus for Economics (STW) is the worldwide largest bilingual vocabulary used for representing and researching economics-related content. It consists of almost 6000 subject headings and more than 20.000 additional entry terms, both available in English and German. It broadly covers topics from the economics domain and other related fields (Kempf and Neubert, 2016). Numerous organizations, libraries, and institutions use the STW for subject indexing and research, e.g., the German Institute for Economic Research.[1] The thesaurus is provided by the Leibniz Information Centre for Economics (ZBW), a large information service provider with the worldwide largest stock of economics literature.[2] The thesaurus is currently maintained manually by a small team of domain experts. They are responsible for deciding whether new terms should be added to the thesaurus, removed, or merged, as well as for finding relationships between those terms. The thesaurus relies on term suggestions from users. To alleviate the task of finding and selecting novel relevant terms manually, we propose a data-driven, automatic way to suggest novel terms for the thesaurus by automatically extracting keywords from domain-specific publications. This approach can not only be used for keyword suggestions for the STW, but also for finding terms for indexing of document collections. We investigate three pre-trained BERT models that are fine-tuned for the task of token classification with the goal to extract domain-specific keywords, which in turn can be filtered to find new suggestions for the thesaurus.

2 Related Work

In recent years, various BERT models have been proposed for the task of keyword and key phrase extraction: Lim et al. (2020) proposed an approach of using two pre-trained BERT models, namely BERT and SciBERT, and fine-tuned them on a task similar to named entity recognition. The former model is pre-trained on the English Wikipedia and the BookCorpus with $3.3B$ tokens (Devlin et al., 2018) and the latter on the Semantic Scholar Corpus with $3.1B$ tokens (Beltagy et al., 2019). For the fine-tuning, each token was assigned to a label, marking either the beginning, middle or end of a key phrase. The models have been evaluated on three different datasets: KDD, WWW and Inspec.[3] KDD consists of abstracts of papers from the ACM conferences on Knowledge Discovery and Data Mining (KDD). WWW consists of abstracts from the World Wide Web Conference (WWW). Both KDD and WWW only include publications between 2004-2014, with 715 and 1330 documents respectively (Gollapalli and Caragea, 2014). Inspec consists of 2000 abstracts of scientific Computer Science journals between 1998 and 2002 (Hulth, 2003). Their reported results show that while their BERT model did not attain state-of-the-art results

[1] https://www.zbw.eu/de/stw-info/anwendungen/
[2] https://www.zbw.eu/en/about-zbw
[3] https://github.com/LIAAD/KeywordExtractor-Datasets

as the maximum performance of their model differs from the state-of-the-art between 0.08 - 5.2%, their SciBERT model overtook the state-of-the-art in all of their datasets with a 3.92 - 8.57% improvement.
Qian et al. (2021) proposed a BERT-based approach for extracting keywords from scientific texts. In their work, BERT is used to extract key sentences from abstracts of papers from the Wanfang database.[4] by dividing abstracts into a set of sentences. For each sentence, BERT is then used to find other sentences with high semantic similarity to the sentence in question. These extracted sentences are then ranked by their similarity and eventually a set of sentences is extracted to further retrieve keywords from. The keyword extraction itself is done by a combination of term frequency-inverse document frequency (TF-IDF) weighting, latent Dirichlet allocation (LDA), and TextRank. The model was evaluated using precision, recall and F1-scores. The results showed an improvement of 1.5% in the F1-score compared to the approach without prior sentence extraction with BERT.
Borisov et al. (2021) also used BERT for keyword extraction by fine-tuning BERT for the task of named entity recognition. They labeled three datasets with a *1* if the word is a keyword, and with a *0* if it is not a keyword. They used two separate datasets, one based on articles from news pages, and one derived from the Qulac datasets for IR-keywords (Aliannejadi et al., 2019). They used two categories for the evaluation of the model: test dataset accuracy and human evaluation. For evaluating the test dataset accuracy they measure precision and recall, as well as the average correct tag identification (ACTI), which tests the overall quality of the assigned tags, e.g., if a word is correctly tagged as a keyword or not, and the correct per response fill (CpRF), which captures the ratio of fully and partially correct predictions. For the human evaluation, a team of human annotators scores each keyword on a score from 1 to 5. The BERT model showed promising results with a precision of 0.86 and a recall of 0.88. The ATCI score measured 0.97, implying that most of the tags have been correctly assigned. The CpRF score of 0.76 implies that two third of the terms have been correctly predicted. The human evaluation score was 3.96, indicating high quality keywords.
In 2022 BERT has been used for domain-specific keyword extraction in combination with an additional Bi-LSTM layer for a sequence labeling task (Pezzo, 2022) fine-tuned on statistics-related textbooks. BERT is used to generate the contextualized word embeddings for the input, which are then fed into a Bi-LSTM layer that helps with the classification of the tokens. Each token is assigned the label "0" if it is predicted as a keyword and the label "0" if not. The difference to the previously presented methods is that this approach is unsupervised, meaning the model has not been trained on labeled texts but on unlabeled texts. The results of the model showed that it performed better than other commonly used keyword extraction methods such as KeyBERT, TextRank, LDA, TF-IDF or TopicRank by a large margin. The model's F1-score was 59.10, whereas the highest F1-score of the compared models was 43.78, obtained by TopicRank.

3 Dataset

In this work, a dataset derived from ECONIS, an online catalogue that contains titles and abstracts from economics literature maintained by ZBW - Leibniz Information Centre for Economics from various economic domains, is used.[5] From the ECONIS dataset, the title, abstract, and metadata of scientific publications are extracted. The full-text body is not used to minimize the complexity of the approaches. The chosen metadata contains the publication year and language of the document. Additionally, three sets of indexing terms are assigned to the publications: assigned by its authors, specialists, and the STW each. Specialists are people from ZBW, that are responsible for subject indexing of documents. They are also responsible for the STW indexing labels, but for that category only terms from the thesaurus can be considered. The dataset is further reduced to publications published between 2009–2021. These restrictions lead to a dataset with 575K entries.

4 Methods

Our approach consists of two steps. First, we fine-tune a BERT model and use it to classify tokens as keyword candidates. Second, we filter the obtained candidates based on frequency and trend.

[4] http://www.wanfangdata.com, accessed 07.07.2023

[5] https://www.econbiz.de/Record/datenbank-econis-online-katalog-der-zbw/10001514790, accessed 18.11.2022

4.1 Extraction Process

To extract domain-specific keywords from documents, three BERT models are fine-tuned for the task of token classification. The first model is SciBERT (Beltagy et al., 2019), which is pre-trained on the semantic scholar corpus. The second model, FinBERT, is pre-trained on financial-communication texts, namely the three financial corpora, *corporate reports 10-K & 10-Q*, *earnings call transcripts* and *analyst reports* (Huang et al., 2022). The third model considered is DistilBERT, which is the lighter version of the original $Bert_{Base}$. It is trained on Wikipedia and a book corpus (Sanh et al., 2020).

To train the models for the downstream task, a labeled dataset is needed. Binary labels are applied to the terms in the documents of the dataset. "1" implies a word is a keyword or part of a key phrase and "0" that the term is not a keyword or part of a key phrase. The labels are assigned to the word based on whether they belong to a term in the STW. Thus, the words of the term "tax consultancy" are each assigned the label "1", however, if the term "consultancy" occurs alone, it is assigned a "0", as it is not an entry in the STW on its own. To fine-tune and then evaluate the models, the dataset needs to be split into training and test set. A subset of the STW terms is randomly sampled and the documents containing any of those terms are assigned to the test set. This ensures that hold-out STW terms have not been seen during fine-tuning. Hereby it can be evaluated how many of these terms that the model has not seen during fine-tuning are predicted as keywords during the evaluation. This subset of terms is referred to as the *control set* and it amounts to 970 terms from which 457 are descriptors and 513 non-descriptors. Descriptors describe the preferred term used for a concept. Non-descriptors describe the same concept, but are secondary terms, e.g., synonyms. The test set thus contains 131K documents and the training set for fine-tuning 443K documents. Each BERT-model variant is fine-tuned for 3 epochs using the training set. The batch size of each model is 32, as recommended by the authors of BERT and the input token length is 512 tokens, the maximal input size for BERT-models (Devlin et al., 2018). The learning rate for fine-tuning is set to $5e-5$.

4.2 Filtering Process

To be able to suggest new terms for a thesaurus, the extracted keywords from the given documents need to be further filtered, because not every extracted keyword is a valuable addition to the STW. The filtering process consists of multiple steps. First, from the pool of extracted keywords, terms are removed that are already part of the STW as well as duplicated terms. This includes singular and plural forms of STW terms.

In the next step, adjectives denoting affiliations to a country are removed, e.g. *French social reform* becomes *social reform*. The adjective makes the term too specific for it to be a relevant term for the STW, considering that the thesaurus needs to be as general as possible. After removing the adjectives, it is verified again whether these terms now belong to an existing entry of the STW, and removed if they do.

The next filter ensures the relevance and frequency of the keyword candidate. Two types of filtering methods are introduced: the frequency filter and the trend filter. The frequency filter considers the frequency of a keyword. If its frequency reaches a threshold, the term is selected as a potential keyword candidate. For the evaluation, a threshold of 300 was chosen. This threshold has been set empirically by analyzing the frequency of existing STW terms during the given time period in the ECONIS dataset. The second filtering method is the trend filter. It selects keywords based on whether their usage has increased in the last three years (between 2019–2021), compared to their frequency in 2009–2018. For this, the average frequencies of those time spans are compared. If the latter average frequency of the term has increased, it is considered as a keyword candidate. Both cases are considered as some terms might not have a high frequency overall, as they have not or barely been mentioned in the literature, but have had a strong increase in recent years, e.g., *Coronavirus* has had a strong increase in recent years for obvious reasons. These terms are just as important as words that are frequent in the literature overall. In the least step of the filtering process, the keyword candidates are standardized to a uniform format, e.g., all candidates are singularized with a capitalized first letter, e.g., *social reforms* becomes *Social reform*.

5 Evaluation

The performance of the proposed models is compared to three common keyword extraction methods: TF-IDF (Luhn, 1957), TextRank (Mihalcea and Tarau, 2004), and KeyBERT (Grootendorst, 2020).

5.1 Term Suggestion

First, each method is evaluated on how effectively it recognizes terms from the control set, thus from the subset of terms that the models have not seen in the fine-tuning phase. Table 1 shows the performance of the methods based on the number of found descriptors (D) and non-descriptors (ND) from the control set in the test set. Besides splitting up the set into descriptors and non-descriptors, each entry for a concept is considered, thus an entry is considered as found by the model if either the descriptor or any of the non-descriptors for this entry are found. An important note to make is that TF-IDF has been given an advantage for this evaluation: because TF-IDF only extracts unigrams from texts but a lot of the terms from the control set and the STW are n-grams, the 10 extracted keywords have been concatenated to one large sequence of terms for each document. It is then evaluated if each subterm of an n-gram occurs in this sequence, if it is the case, then the term is considered as found. If only a subset of words of the term has been found, then the term is not considered as being found. In practice, it would not be known what terms are expected to be found, thus every combination of the extracted terms would have to be considered.

Beginning with the results for the descriptors, TF-IDF has in fact found 100% of the descriptors with its given advantage. Aside from that, DistilBERT performed the best by finding about 84% of the descriptors in the control set. This leaves a 20% margin compared to the next best method, which is TextRank. However, the two remaining fine-tuned models SciBERT and FinBERT show worse results than DistilBERT and TextRank. The results for the non-descriptors show that this time TF-IDF only finds about 13% of the non-descriptors, thus performing the worst out of all evaluated methods. Again, DistilBERT shows the best performance by finding 61% of the non-descriptor terms, which shows a 20% increase compared to the results of TextRank once again. Hence, counting an STW entry as found if either the descriptor or any of the non-descriptors are found, TF-IDF results in finding 100% of the entries, due to its performance on the descriptors. DistilBERT extracts terms for nearly 91% of the entries from the control set, given its performance on both the descriptors and the non-descriptors. This shows that the DistilBERT model works well in finding new and domain-specific keywords from documents. However, SciBERT and FinBERT do not show promising results.

Table 1: Percentage of found terms from the control set in the test set

	D	ND	Both
DistilBERT	83.6%	**61.0%**	90.8%
SciBERT	55.6%	27.3%	68.1%
FinBERT	50.5%	24.0%	60.8%
KeyBERT	35.4%	24.8%	46.0%
TextRank	63.7%	41.5%	76.2%
TF-IDF	**100.0%**	12.7%	**100.0%**

Besides the performance on the control set, it is also interesting how the extracted keywords compare to the labels assigned to the documents in the dataset, thus how many of the STW terms have been extracted as keywords by the methods. Therefore, precision, recall, and F1-scores are calculated for every method. Precision describes how many of the retrieved keywords are marked as keywords in the labeled dataset, while recall determines how many of the overall keywords have been retrieved (Roelleke, 2013). Table 2 shows these results when considering terms that have been retrieved only partially, as each term has its own label. With these measures, it can be evaluated how well the proposed models and other keyword extraction techniques can recognize the terms that are part of the STW. Based on these values, all proposed BERT models outperform the baseline methods by a large margin. SciBERT, FinBERT and DistilBERT have each resulted in precision and recall values higher than 94%. These values are very high, which is likely due to the fact that these models have been trained on documents containing a large amount of STW terms. Hence they are much more likely to extract these terms as keywords. The other methods lack the domain-expertise as they have not been trained on the same data. Aside from these models, TF-IDF (Luhn, 1957) performed the best from the baseline methods, but it only reached values of up to 44%, thus resulting in a large margin compared to the fine-tuned BERT models. This shows the advantage of training a keyword extraction model

Table 2: Comparison of the extracted keywords with the labeled test set

	Precision	Recall	F1
DistilBERT	**0.97**	**0.99**	**0.98**
SciBERT	**0.97**	0.97	0.97
FinBERT	0.94	0.94	0.94
KeyBERT	0.28	0.22	0.25
TextRank	0.43	0.33	0.38
TF-IDF	0.44	0.41	0.42

on a domain-related dataset, as it is familiar with terms that it it has seen during pre-training.

5.2 Manual Evaluation

To suggest new terms for the STW, the extracted keywords and key phrases have been run through the filtering process, filtering out terms that are already part of the STW and then applying either a frequency (FF) or trend filter (TF). The threshold of the frequency filter is set to 300. Then for each keyword extraction method and filter type, 100 terms have been randomly selected from the pool of keywords. Each of these sampled keywords is then presented to an expert from the STW team for evaluation. Based on the performance of the three proposed BERT models in the prior experiments, DistilBERT is selected to be further evaluated manually together with the baseline methods. All of the terms, in total 800, are then combined into one randomly sorted list and are presented to the STW team member along with the frequency of the suggested term. For each term, the STW member then labels the keyword with "1", if he/she thinks that the term has the potential to be added to the STW as either a descriptor or a non-descriptor, and label "0" if it is not a fitting word for the STW.
Table 3 shows the precision results of the manual evaluation for each filtering type. For the keywords that have been selected based on their frequency, the baseline methods TextRank and TF-IDF did not perform well. TF-IDF actually performed the worst on both filter types, having only 17 frequency-based keywords selected as potential keywords and 31 terms for the time filter (out of 100). TextRank performed slightly better than TF-IDF but worse than the other methods. While for KeyBERT 44 out of 100 terms have been marked as potential keyword candidates, DistilBERT found even more, resulting in 51% of the suggested terms being potential keywords

Table 3: Precision after frequency filtering (FF) and trend filtering (TF)

	FF	TF	Overall
DistilBERT	**51%**	**59%**	**55%**
TextRank	22%	42%	32%
TF-IDF	17%	31%	24%
KeyBERT	44%	36%	40%

for the STW. The DistilBERT model performs even better for trend-filtered keywords. 59 of the 100 selected terms qualify as potential keywords for the STW. The model outperforms the baseline methods by a large margin of 17%. The second-best performance shows TextRank, which still only suggested 42 potential keywords. The table also shows the overall percentage of terms that can be considered as potential candidates for the STW. The results show that the DistilBERT model suggests the best keyword candidates for the STW. More than 55% of the suggested terms qualify as potential candidates for addition to the thesaurus. Compared to the baseline methods, our model showed an increased performance of 15%.

These results also show that for 3 out of 4 applied keyword extraction methods, the trend filtering resulted in more potential keywords than the frequency filter.

5.3 Document Indexing

Next up, we evaluate whether the extracted keywords from the different methods can be used to index documents. Based on the performance of the proposed models on their ability to extract a significant portion of the STW terms, they might be able to produce indexing terms for documents directly. Thus, we analysed how many of the extracted keywords correspond to indexing terms from any of the three label sets: STW labels, author labels, and specialist labels, as described in Section 3. While only for a small portion of the dataset these indexing terms are provided, it can at least be evaluated whether the models extract these existing terms. Hence it would be even more useful if this model predicts the labels well enough to be used for automating the labeling of documents. Unfortunately, only around 126K of the 575K entries of the entire dataset are indexed with any of the terms from the three index labeling sets, resulting in only around 22%. For the test set, only 1.3% of the documents

Table 4: Available indexing labels in the test set

Indexing Set	Available Labels
STW	3345
Author	435
Specialist	36

Table 5: Percentage of extracted keywords corresponding to document labels

	STW	Author	Specialist
DistilBERT	**91.3**%	**34.9**%	**27.8**%
SciBERT	85.0%	33.1%	0.0%
FinBERT	75.0%	32.0%	19.4%
KeyBERT	48.5%	21.4%	25.0%
TextRank	25.2%	11.3%	11.1%
TF-IDF	29.6%	12.6%	8.3%

contain any indexing terms in the metadata. Table 4 lists the labels in the test set for the different indexing sets.

Table 5 shows the number of labels that have been correctly predicted by the keyword extraction methods. Overall, in each of the label categories, our DistilBERT model performed the best by finding the largest number of labels each. For the STW Labels, the DistilBERT model correctly predicted approximately 91% of the given labels. For the baseline methods, KeyBERT performed the best, but only extracted around 48% of the labels. The results are similar for the author labels: While the DistilBERT model only predicts around 35% of the labels this time, it still performed better than the baseline methods, from which KeyBERT performs the best again with 21% of found labels. For the specialist labels, only 36 labels were available in the test set. While DistilBERT performs the best again by predicting 28% of the labels, it did not perform better by a large margin compared to the other methods this time, as the performance of KeyBERT comes close with 25%. Following these results, our DistilBERT model performs the best in finding labels for documents. Especially in the case of the STW labels our model may be useful, as these results suggest that it finds the correct words in documents. Considering the fact that only a small amount of texts have any labels available, it might be worth using this model to suggest indexing terms for documents.

6 Discussion

Analyzing the keywords extracted by either of the methods together with comments from the domain expert, some common errors from the methods can be identified. One of the occurring problems relates to the part-of-speech of the extracted keywords. The STW only accepts entries of nouns, not verbs or adjectives, which have been commonly extracted by all of the methods. This can be improved by implementing an additional part-of-speech filter in the filtering process to only consider nouns as candidates for the STW. A similar problem occurs with the extraction of proper names and corporation names. These are terms that are not considered for the STW, but at this point, the proposed model does not recognize them and thus also not remove these terms from the candidate pool. The results in the previous section suggest that the fine-tuned DistilBERT model can be used to label documents with indexing terms from the STW. Given the fact that all three of the proposed models are fine-tuned the same way, it can be presumed that the increased performance of BERT relates to the pre-trained model itself, thus the corpus of the DistilBERT model appears to create the best-fitting model for this use case. This is supported by the fact that SciBERT as well as FinBERT in multiple cases did not know a token, thus labeling them with the as [UNK]. However, since only a small part of the test set had been labeled at all, the experiment should also be carried out on a larger set of indexed documents, e.g., the complete dataset. Furthermore, the methods predict more keywords for a document than the number of indexing terms available for each document. Therefore it would be beneficial to rank candidates from a document and only suggest the most important ones. For future work, a way of building an actual term hierarchy could be considered, making use of hierarchical connections among thesaurus terms. While first experiments on clustering terms did not show promising results, finding a way to not only grouping terms but also determining the descriptor terms would be helpful.

7 Conclusion

In this work, the three pre-trained BERT models DistilBERT, SciBERT, and FinBERT were fine-tuned for the task of token classification with the goal of domain-specific keyword extraction. Their performance has been compared to three baseline methods used for keyword extraction, namely TF-

IDF, TextRank and KeyBERT. The results showed that DistilBERT performed the best overall, as it was able to extract domain-specific keywords reliably, but also to suggest more potential new terms for the Thesaurus for Economics (STW) compared to the other methods. This suggests that fine-tuning a model on domain-related documents does indeed help in retrieving domain-specific terms compared to not fine-tuned methods. In future research, the filtering process could be further optimized to achieve higher precision by limiting the number of suggested terms.

References

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. In *EMNLP*. Association for Computational Linguistics.

Oleg Borisov, Mohammad Aliannejadi, and Fabio Crestani. 2021. Keyword extraction for improved document retrieval in conversational search.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristin Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding.

Sujatha Das Gollapalli and Cornelia Caragea. 2014. Extracting keyphrases from research papers using citation networks. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 1629–1635. AAAI Press.

Maarten Grootendorst. 2020. Keybert: Minimal keyword extraction with bert.

Allen H. Huang, Hui Wang, and Yi Yang. 2022. Finbert: A large language model for extracting information from financial text. *Contemporary Accounting Research*.

Anette Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, EMNLP '03, page 216–223, USA. Association for Computational Linguistics.

Andreas Oskar Kempf and Joachim Neubert. 2016. The role of thesauri in an open web: A case study of the stw thesaurus for economics. *Knowledge Organization*, 43:160–173.

Yeonsoo Lim, Deokjin Seo, and Yuchul Jung. 2020. Fine-tuning bert models for keyphrase extraction in scientific articles. *Journal of Advanced Information Technology and Convergence*, 10(1):45–56.

H. P Luhn. 1957. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1:309–317.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *EMNLP*, pages 404–411. Association for Computational Linguistics.

Lorenzo Pezzo. 2022. Keyed alike: Towards versatile domain-specific keyword extraction with bert. Master thesis. Utrecht University.

Yili Qian, Chaochao Jia, and Yimei Liu. 2021. Bert-based text keyword extraction. *Journal of Physics: Conference Series*, 1992(4):042077.

Thomas Roelleke. 2013. Information retrieval models: Foundations and relationships. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 5.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.