DEIM2022 E41-3(day4 p10)

Topic Modeling using Jointly Fine-tuned BERT for Phrases and Sentences

Zikai ZHOU† and Kei WAKABAYASHI††

† College of Knowledge and Library Sciences, School of Informatics, University of Tsukuba
1–2 Kasuga, Tsukuba-shi, Ibaraki 355–8550, Japan
†† Faculty of Library, Information and Media Science, University of Tsukuba
1–2 Kasuga, Tsukuba-shi, Ibaraki 355–8550, Japan
E-mail: †s1813024@s.tsukuba.ac.jp, ††kwakaba@slis.tsukuba.ac.jp

Abstract Using topic modeling to analyze large collections of documents has been a functional approach for many years. However, traditional topic models such as Latent Dirichlet Allocation rely on the assumption of "Bag-of–Words", which ignores the connection and inner semantics between words in terms of phrases. Although phrases act as important grammatical units in human language, due to the restriction on vocabulary size and model complexity, less research has been conducted on phrase-level topic models. In this research, we propose a phrase-level topic model based on pre-trained distributed representations of words, documents, and phrases. We build this model based on the Top2vec topic model and BERT embeddings. Due to the fact that there is no existing BERT-based model designed to produce embedding for both sentences and phrases, we propose a jointly fine-tuned BERT model for sentences and phrases embeddings. Our experiments demonstrate that the jointly fine-tuned BERT could produce high-quality sentences and phrases embeddings, and the model could generate well-performed phrase-level topics.
Key words topic model, fine-tuning, BERT, phrases, semantic space

1 Introduction

Natural Language Processing (NLP) is a subfield of computer science concerned with the interactions between computers and human language. Given an NLP model, we want the computer could somehow "understand" the context, and then extract information to achieve tasks such as semantic analysis or text classification. One main problem of natural language processing is to deal with a large collection of text which cannot be reasonably read and sorted by humans. We may want to discover the theme or subject from a text, and then organize the whole collection automatically.

Topic models are designed to achieve this goal. A topic is considered as a latent semantic structure of a text, which could tell us what is the text talking about. Topic models are used to discover topics from a collection of documents and then be used to summarize documents, search documents by queries or keywords, or categorize documents by topics. Traditional unsupervised topic models are based on the Bayesian probabilistic model. Models such as Latent Dirichlet Allocation (LDA) [1] and Probabilistic Latent Semantic Analysis (PLSA) [2] discover topics by finding statistical features of words. Some modern topic models such as Neural Variational Document Model (NVDM) [3] and Document Neural Autoregressive Distribution Estimation (DocNADE) [4] use neural networks to generate more expressive topics.

One of the major issues in topic modeling is handling phrases. A phrase is a certain group of words that act together as a grammatical unit. In both syntax and grammar meaning, the combination of certain words could lead to a different purpose. In general, phrases could carry more information than single words. Some phrases could have meanings that cannot be determined by any of their components, and some words are even meaningless without related words. In topic modeling, phrases are likely to lead to ambiguity if we only consider words as grammatical units. Words such as *white* and *house* appearance may lead to an architecture-related document, while the phrase *White House* appearance leads to a politics-related document. Many traditional topic models such as LDA are based on the popular assumption of Bag-of-Words (BOW), in which the ordering-based semantics of words are ignored. There is some research such as Bigram Topic Model (BTM) [5], Topical N-gram Model (TNG) [6] and Phrase-based LDA [7] which consider phrases by simply discovering and adding phrases into the original vocabulary set, and learning those phrases just like normal words. This approach however would make an enormous vocabulary set, then lead to problems such as huge models and

unstable learning. These problems become even more serious on some well-performed neural-network-based models.

An alternative way of topic modeling is to use distributed representations of words and documents. Learning embeddings for representations of words and documents has been a popular field in NLP. This method is behind the idea of *distributional hypothesis*, in which John Rupert Firth famously said "You shall know a word by the company it keeps" [8], indicates that similar words are often used in similar contexts. Word2vec [9] firstly introduced the continuous skip-gram model and used it to capture distributed word representations, while later models such as GloVe [10] and Bidirectional Encoder Representations from Transformers (BERT) [11] produced even better word embeddings using different methods. Compared to traditional approaches, each of them produced state-of-the-art results on many NLP tasks. Top2vec [12] introduced a new way of topic modeling by using jointly embedded documents and word vectors. After creating semantic embeddings for both documents and words, Top2vec applies dimensionality reduction and clustering to document embeddings and generates topic vectors in the same semantic space. This method provides benefits such as automatically finding the number of topics and finding more informative and representative topics of the corpus over traditional topic models such as LDA and PSLA.

The original paper of Top2vec chose Doc2vec [13] to embed documents and words into the same semantic space. However, using pre-trained encoding models such as Universal Sentence Encoder (USE) [14] or BERT could also be helpful when it comes to efficiency and performance on small datasets. Pre-trained models are often trained on massive datasets and could produce high-quality embeddings for downstream tasks. While USE is pre-trained to produce sentence embeddings, BERT is only pre-trained for word embeddings originally. An alternation of BERT called Sentence-BERT [15], which is fine-tuned to encode sentences, could be used instead to embed documents and words into the same semantic space.

To conduct a phrase-level Top2vec model, phrases should also be embedded into the same semantic space as well as words and documents. Phrase-BERT [16], which is based on the Sentence-BERT architecture, is a fine-tuned BERT-base model that shows state-of-art results on phrase embeddings. However, after the fine-tuning for phrases, the model lost its ability to embed documents and create meaningful clusters. To solve this problem, we propose a joint fine-tuned BERT model by training the BERT model with mixed datasets and conducting a phrase-level topic model which generates phrase-level topics based on semantics. Our experiments show that the jointly fine-tuned BERT could produce both sentences and phrases embeddings for the topic model, and the model could generate better phrase-level topics.

2 Related Work

2.1 Topic models

LDA is a widely used generative statistical topic model in natural language processing that describes each document as a distribution of topics, and each topic as a distribution of tokens. It is a generalized form of PLSA which adds a Dirichlet prior distribution on document-topic and topic-word distributions. Based on the idea of topic distributions, different generative topic models such as Correlated Topic Model (CTM) [17] and Structured Topic Model (STM) [18] are proposed. Some other approaches such as NVDM and DocNADE use modern neural networks to generate similar distributions. However, these models are all based on the Bag-of-Words assumption, and therefore the ordering and combination of words are ignored.

Beyond the Bag-of-Words assumption, some research conduct n-gram models with existing topic models. BTM firstly introduce bigram tokens into LDA models, while TNG use a combination of unigrams and bigrams for longer phrase tokens. Due to the limitation of vocabulary size, these models are limited to bigram models. Other models constrain phrases to limit the size of the vocabulary. Yu et al. [7] conduct a C-value for phrase importance and apply a threshold onto the topic model, while Li et al. [19] integrates a regular expression constraint condition to filter phrases. Despite these models successfully introducing meaningful phrases into topic models, the constraint ignores the semantics of phrases and may discard some informative phrases.

Top2vec is a topic model which leverages joint document and word semantic embedding to find topic vectors. By using distributed representations of words and documents, it overcomes the weaknesses of BOW representations of documents in terms of latent semantics. While the original model learns a Doc2vec model for embeddings, the model also shows the potential of being a fast model by directly using pre-trained models such as USE and BERT. Our proposed model is a phrase-level extension based on the Top2vec model using pre-trained BERT models.

2.2 Embeddings

Our work relates to the idea of learning dense representations for semantic units, particularly on the modern neural-network-based pre-trained models. Word2vec introduced Skip-gram and C-BOW models for learning word embeddings, and further research [20] shows its potential for phrase embeddings. The idea is also implemented by different models such as GloVe and fastText [21]. Furthermore, Doc2vec, USE, and Embeddings from Language Models (ELMo) [22]

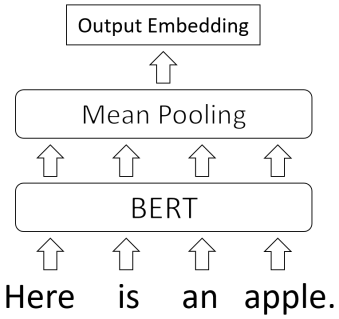

Figure 1 Basic structure for BERT embeddings. Although not shown in the graph, [CLS] and [SEP] tokens will be added to the begin and end of the sentence.

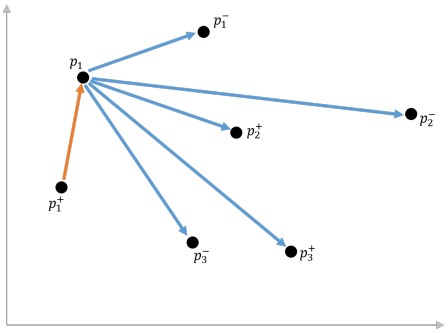

Figure 2 Multipel Negative Ranking Loss with hard negatives. As shown, p_1 is pulling towards p_1^+ and pushing away from p_2^+, p_3^+, p_1^-, p_2^-, and p_3^-.

use different approaches for generating sentence embeddings.

The advent of huge-scale pre-trained language models such as BERT opens a new way for word and sentence embeddings. It is shown that the pre-trained BERT model benefits many neural language processing tasks [23], while the modified Sentence-BERT shows its ability to produce high-quality embeddings for unsupervised tasks such as text comparison. However, Yu and Ettinger [24] show that BERT struggles to produce meaningful embeddings for short semantic units such as words or phrases. Phrase-BERT proposes a phrase-specific version of BERT and shows that it produces meaningful phrase embeddings while also promoting a lexically diverse semantic space. To achieve our goal of generating joint document, phrase, and word embeddings for topic modeling, we propose a joint fine-tuning process based on Sentence-BERT and Phrase-BERT.

3 Proposal

We propose a joint fine-tuning task on top of BERT for both sentences and phrases, which relies on contrastive objectives similar to Sentence-BERT. The training dataset contains triplets for both sentence and phrase.

3.1 BERT Embeddings

For sentence and phrases embeddings, we follow the procedure of the Sentence-BERT model [15], which takes the mean of all output vectors as the final embedding. Given an input X of length N tokens, by adding a mean pooling layer on top of the BERT model, the final representation $E(X)$ could be computed as:

$$E(X) = \frac{\sum_{i=1}^{N} \text{BERT}(X_i)}{N},$$

where $\text{BERT}(X_i)$ indicates the final-layer token-level output for token X_i. This structure is depicted in Figure 1.

3.2 Fine-tuning Approach

We use the triplet network [25] to update the model weights so that the embeddings produced by the model are meaningful and could be compared with cosine-similarity. Consider a sentence or phrase as a semantic unit, the network is trained on training data tuple which contains an anchor unit p, a positive example p^+, and a negative example p^-. We assume that in general, p and p^+ will have a similar meaning, while p and p^- will have a different meaning. The model is encouraged to produce close embeddings for p and p^+ while pushing the embeddings for p and p^- apart.

The network uses Multiple Negative Ranking Loss [26] as the loss function for the weight update. Given a single training data batch as

$$(\mathbf{p}, \mathbf{p}^+, \mathbf{p}^-) = \{(p_1, p_1^+, p_1^-), (p_2, p_2^+, p_2^-), ..., (p_n, p_n^+, p_n^-)\},$$

we compute the similarity between two units, for example p and p', as $S(p, p')$. Specifically, the cosine-similarity between two embeddings are used for the calculation of similarity.

$$S(p, p') = \cos\left(E(p), E(p')\right) = \frac{E(p) \cdot E(p')}{\|E(p)\| \|E(p')\|}$$

For the efficiency of the model, we treat the corresponding positive example p_i^+ as positive, and all other positive examples $p_j^+ (j \neq i)$ as negative. In addition, since we use triplet for training, all negative example p_n^- would also be considered as negative. The embedding will be pulling towards embedding of the positive example and pushing away from embeddings of negative examples, as shown in Figure 2. The cross-entropy loss would then be computed as:

$$J(\mathbf{p}, \mathbf{p}^+, \mathbf{p}^-) = -\frac{1}{n} \sum_{i=1}^{n} [S(p_i, p_i^+) - \log \sum_{j=1, j \neq i}^{n} e^{S(p_i, p_j^+)} - \log \sum_{k=1}^{n} e^{S(p_i, p_k^-)}],$$

and would be minimized through back propagation.

3.3 Sentence Training Data

Following the Sentence-BERT [15] settings, the sentence training data is the combination of the SNLI [27] and the Multi-Genre NLI [28] dataset. These two datasets contain

sentence pairs and one of the three labels: *entailment*, *neutral*, and *contradiction*. The pairs labeled as *entailment* would be considered as positive pairs and the *contradiction* would be added as hard negatives.

3.4 Phrase Training Data

Following the Phrase-BERT [16] settings, the phrase training data contains two different objectives for high-quality phrase embeddings.

The first objective mainly aims at lexically diverse phrasal paraphrases. Top 100K phrases are extracted from the WikiText-103 Corpus [29] using the SR-parser from CoreNLP [30]. Given the phrase p, positive example p^+ is created by passing p through the GPT2-based diverse paraphrasing model [31] and decoding the using nucleus sampling with the nucleus probability mass of 0.8 [32] with lexical constraints. Negative sample p^- is created by passing a randomly sampled phrase to the paraphraser.

The second objective mainly aims at phrases in context. Top 100K phrases of length less than 10 tokens are extracted from the Books3 Corpus [33] along with its context of length 120 tokens. Given the phrase p, positive example p^+ is created by replacing the occurrence of p within the context with a `[MASK]` token, while p^- is a randomly sampled context from the corpus.

3.5 Topic Model

Our proposed phrase-level topic model is based on Top2vec [12] model structure. The Top2vec model generates topic, document, and word vectors that are all jointly embedded, with the distance between them representing semantic similarity. While the origin Top2vec model uses Doc2vec embeddings which are trained from the corpus, our proposed model uses a pre-trained BERT model for embeddings. Here is the basic procedures of the phrase-level topic model.

（1） Mine NP, VP, ADJP, and ADVP phrases of length between 2 to 10 words through CoreNLP SR-parser.

（2） Create embeddings for words, phrases, and documents using pre-trained BERT models.

（3） Reduce dimensions of document embeddings using Uniform Manifold Approximation and Projection for Dimension Reduction (UMAP) [34].

（4） Find clusters of document embeddings using Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) [35].

（5） Find centroid vectors of document clusters in original dimension as topic vectors.

（6） Assign n-closest words to the topic vectors as topic words.

3.6 Implementation Details

We fine-tune the joint model on two NVIDIA GTX 1080ti GPU parallelly for 1 epoch. A batch size of 16 is used along with a learning rate of 2e−5 with Adam [36]. Following the linear warm-up setup of Sentence-BERT [15], the initial 10% of training steps are used as warm-up steps. For comparison, three models are separately trained based on BERT, Sentence-BERT, and Phrase-BERT with the same training setups and datasets, named as `joint-BERT-based`, `joint-Sentence-BERT-based`, and `joint-Phrase-BERT-based` respectively. For phrase-level topic model, following the default setup [12], we use the UMAP as the dimensionality reduction tool and HDBSCAN as the clustering tool based on Python implementations. For UMAP parameters [37], we use `n_neighbors=15` and `n_components=25`. For HDBSCAN [38], we use `leaf` as the cluster selection method.

Dataset	Example pair	Score
STSB	(The bird is bathing in the sink., Birdie is washing itself in the water basin.)	5.0
Turney	(learned person, pundit)	match
BiRD	(business development, economic growth)	0.586

Table 1 Datasets and example pairs used in our embedding evaluation.

4 Experiment

In this part, we show the performance of our proposed models through several experiments. We compare our three models with BERT, Sentence-BERT, and Phrase-BERT as baselines. Since the research [15] shows that using the mean-pooled representation over the final-layer outputs outperforms other methods such as using the `[CLS]` representation, a mean-pooling layer is added to the original BERT model (other models have already implemented this feature).

4.1 Embedding Evaluation

We evaluate both sentence and phrase embeddings on different tasks (Table 1).

For sentence embeddings, we use the common Semantic Textual Similarity (STS) tasks, in specific the STS-benchmark [39] dataset. STS datasets provide a series of sentence pairs and labels between 0 and 5 on their semantic relatedness. Following the setup of Sentence-BERT [15], with no STS-specific training, the Spearman's rank correlations between the cosine-similarity of the sentence embeddings and the gold labels are computed for comparison.

For phrase embeddings, two different phrase-level semantic relatedness tasks are used following previous works on evaluating phrase embeddings. Turney [40] contains groups of five unigrams and a corresponding bigram. The model is asked to tell which of the five unigrams has the closest meaning with the given bigram. The cosine-similarity of unigram and bigram embeddings is computed and compared to

Model	STSB	Turney	BiRD
Baseline Models			
BERT	0.4729	0.4261	0.4437
Sentence-BERT	**0.8505**	0.5183	0.6869
Phrase-BERT	0.7782	0.5720	0.6880
Proposed Models			
joint-BERT-based	0.8220	0.5789	**0.7086**
joint-Sentence-BERT-based	0.8274	**0.5803**	0.7063
joint-Phrase-BERT-based	0.8316	0.5638	0.7043

Table 2 Results on embedding evaluation.

find the answer. The final accuracy is used for comparison. BiRD [41] is a correlation task consisting of pairs of bigram phrases and a human-rated similarity between 0 and 1. The Pearson correlation coefficient between the gold label and the cosine-similarity of the bigrams is computed for comparison.

The results are shown in Table 2. For the STS sentence embedding evaluation, the best score among the joint model is 0.8316, while the Sentence-BERT produces the overall best score of 0.8505. As the performance is slightly worse compared to the Sentence-BERT, all three joint models give a significant improvement compared to the Phrase-BERT score of 0.7782. For the phrase level Turney task, two of the joint models produce higher scores compared to the baseline phrase model Phrase-BERT, given that the best score is 0.5803 compared to the Phrase-BERT score of 0.5720. For the phrase correlation task of BiRD, all three joint models produce significantly higher scores compared to the baselines, given that the best score is 0.7086 compared to the best baseline score of 0.6880.

4.2 Topic Model Evaluation

Following the procedure of section 3.5, we construct the phrase-level topic model using three baseline models and three proposed joint BERT models. We choose the well-known 20 News Groups Dataset [42] as the training datasets for topic model evaluation. While the 20 News Groups Dataset contains 18,831 posts with labels they are posted in, none of the gold labels are used in the training of the topic model. The phrases are extracted in advance and added to the vocabulary. In addition, we set frequency thresholds for the frequency of words and phrases in order to avoid unstable results. While documents would be directly embedded by the embedding model, only words appearing more than 25 times and phrases appearing more than 10 times are embedded for topic word representation.

For topic evaluation, we use UMass topic coherence [43] metric. It is shown that UMass topic coherence conducts similar results with the human rating for topic quality, and implements the idea of similar words are often used in similar contexts. Given topic t belongs to the set of all topics

Model	Topic Number	Coherence	Phrase Coherence
Baseline Models			
BERT	10	-586.87	-138.38
Sentence-BERT	**85**	-549.60	-135.71
Phrase-BERT	11	-523.00	**-109.53**
Proposed Models			
joint-BERT-based	**90**	-504.12	-128.27
joint-Sentence-BERT-based	**88**	-494.33	-128.84
joint-Phrase-BERT-based	**91**	**-494.13**	-131.08

Table 3 Results on topic evaluation. For coherence score, top 20 words of each topic are used. For phrase coherence, top 10 phrases of each topic are used.

$T = \{t_1, t_2, ..., t_N\}$, the UMass topic coherence for topic t could be computed as follow:

$$C(t; V^{(t)}) = \sum_{m=2}^{M} \sum_{l=1}^{m-1} \log \frac{D(v_m^{(t)}, v_l^{(t)}) + 1}{D(v_l^{(t)})},$$

where $V^{(t)} = \{v_1^{(t)}, v_2^{(t)}, ..., v_M^{(t)}\}$ denotes a list of the top M words for topic t, $D(v)$ denotes the document frequency of the word v (i.e., the number of documents which v appears at least once), and $D(v, v')$ denotes the co-document frequency of the words v and v' (i.e., the number of documents which both v and v' appear at least once). To prevent calculating the logarithm of 0, a smoothing count of 1 is provided. Since more related words appear in the same topic means better the topic concludes a subset of documents, a higher coherence score refers to higher topic quality.

The average of each topic's coherence would be served as the final coherence score of the topic model:

$$C(T; V) = \frac{1}{N} \sum_{i=1}^{N} C(t_i; V^{(t_i)}),$$

where N is the number of topics and is automatically found by the topic model, thus models that produce different numbers of topics could be compared.

In the experiment, we employ the top 20 topic words which contain phrases for each topic. The overall coherence scores along with the topic number found by each model are displayed in Table 3. In addition, in order to see how well phrases alone perform in topics, we calculate the coherence scores for the top 10 phrases for each topic as well. For topic numbers generated by each model, the BERT and the Phrase-BERT model produce significantly lower numbers as 10˜, while the Sentence-BERT and all three joint BERT models produce more than 80 topics. For the overall coherence score, the baseline models give the best score of -523.00 produced by Phrase-BERT. While all three joint models produce better scores compared to baseline models, the best

Model	Overlap Rate
BERT	0.024
Sentence-BERT	0.034
Phrase-BERT	0.019
joint-BERT-based	0.029
joint-Sentence-BERT-based	0.034
joint-Phrase-BERT-based	0.030

Table 4 Lexical overlap rates for each model in top 20 topic words.

score given by the proposed models is -494.13. For the coherence score of the top 10 topic phrases, the best score is produced by the Phrase-BERT model of -109.53. The joint models outperform the BERT and the Sentence-BERT model with the best score of -128.27.

5 Discussion

According to the result, we show that our joint model could produce high-quality embeddings for both sentences and phrases. Since STS data is included in the training data for Sentence-BERT, it yields the best score among all models. Phrase-BERT loses some of its ability to produce sentence embeddings since there is no sentence-targeting data involved in its training dataset. Besides, by adding NLI data to the training dataset, the proposed joint model successfully maintains its ability to produce sentence embeddings, no matter which base model is chosen for the last-step fine-tuning. For phrase embeddings, the joint models outperform not only Sentence-BERT but also Phrase-BERT, which means that the sentence training data may also improve the quality of phrase embeddings when the correct proportion is mixed with phrase training data. Specifically, fine-tuning based on BERT or Sentence-BERT seems to produce better results compare to Phrase-BERT.

Furthermore, the topic evaluation results show that the joint model does improve topic quality in terms of phrase-level topic modeling. Since the topic model conducts clustering on document embeddings, the model's ability of embedding sentences is highly related to the topic number. Due to the fact that BERT and Phrase-BERT have not conducted specific training for sentences embeddings, the document embeddings are mixed together, thus few topics are discovered by the model. Sentence-BERT as well as the proposed models on the other hand produce quality sentence embeddings for clustering and generate reasonable numbers of topics from the corpus. The coherence scores show that under similar circumstances, the joint models outperform Sentence-BERT in terms of topic quality. We also observe the relationship between phrases and topics with phrase coherence scores. The results show that Phrase-BERT has an advantage due to the fact that it mainly targets phrase embeddings. Nevertheless, the joint model outperforms other baseline models, proving that it successfully obtain the ability to produce joint embeddings for documents and phrases through joint fine-tuning. Overall, the results meet our expectations on the jointly fine-tuned BERT models.

In addition, we take a deeper look at our proposed topic model. One of the big differences between other phrase-level topic models [5] [6] [7] [19] and our proposed model is that instead of tokenized corpus to the combination of phrases and words, our proposed model directly mines phrases and adding them into the vocabulary. While having the benefit of not restricting the type and form of phrases, this approach does bring the concern of lexical overlaps in words and phrases. Since we use the UMass coherence score to evaluate topics that involve the counting of co-document appearance, there is the possibility that more lexical overlaps in topic words produce a better coherence score. Therefore, we calculate the percentage of overlapping word pairs in all topics, which is shown in Table 4. The results show that based on the structure of mean-pooling BERT outputs, there are actually few lexical overlapping appearances in topic words, which brings minimal influence to the final result.

One of the goals of our proposed topic model is to build a phrase-level topic model which relies on semantic similarity. Although it is proved that BERT-based models have advantages in terms of semantic interpretation [11], we still want to know how well they perform in the phrase-level topics. Thus, we generate Wordclouds for the top 20 topic words of some presentive topics, and the result is shown in Figure 3. It is easy to tell that these topics relate to religion, medicine, cryptology, sports, politics, and spaceflight respectively. Proper nouns such as *NASA*, *NHL*, *NSA* and phrases such as *the White House*, *the Stanley Cup*, *copy protection*, *the space program* are successfully embedded into the correct topics, meaning that the joint model captures the semantics of those words and utilizes them into the topic model, which traditional statistical topic models failed to achieve.

The future challenges for our research focus on improving the quality and enhancing the topic model structure. Since the main goal of our proposed joint model is to embed sentences, words, and phrases into the same semantic space, we choose the joint fine-tuning approach based on two previous works [15] [16]. There are other approaches worth trying such as using different tagging tokens for phrases and words, training the model with more and better datasets that offer meaningful hard negatives, and using advanced base models such as RoBERTa [44] and MPNet [45]. Besides topic modeling, there are still plenty of possibilities for the joint BERT-based language model. For topic modeling, we expect the proposed

Figure 3 Wordclouds for top 20 topic words of topic No.8, No.24, No.35, No.46, No.55, No.58 from `joint-BERT-based` topic model.

model to show effectiveness on datasets other than 20 News Groups Dataset. Moreover, the problem of unstable topic numbers remains. Although the model could automatically find the topic number, the number is still related to hyperparameters of HDBSCAN and UMAP. The future challenge would be using prior knowledge or topic-specific fine-tuning to enhance our topic models.

6 Conclusion

In this research, we proposed a jointly fine-tuned BERT model for phrase and sentence embeddings and utilized it on a phrase-level topic model. We showed that previous models focus on either sentence or phrase embeddings, which is not optimal for topic models based on the semantics of these different grammatical units. Our proposed model took advantage of the joint fine-tuning process and performed well on both sentence and phrase embedding evaluations. Different from traditional statistical topic models, the proposed topic model finds topics based on the semantics of documents, phrases, and words. Furthermore, by conducting topics from different kinds of BERT-based models and comparing the result on topic coherence, we showed that our proposed topic model successfully generated phrase-level topics and outperformed other baseline models.

References

[1] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, Vol. 3, pp. 993–1022, 2003.

[2] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '99, pp. 50–57, New York, NY, USA, August 1999. Association for Computing Machinery.

[3] Yishu Miao, Lei Yu, and Phil Blunsom. Neural variational inference for text processing. November 2015.

[4] Stanislas Lauly Hugo Larochelle. A neural autoregressive topic model. In *Advances in Neural Information Processing Systems 25*.

[5] Hanna M Wallach. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pp. 977–984, New York, NY, USA, June 2006. Association for Computing Machinery.

[6] Xuerui Wang, Andrew McCallum, and Xing Wei. Topical N-Grams: Phrase and topic discovery, with an application to information retrieval. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pp. 697–702. ieeexplore.ieee.org, October 2007.

[7] Zhiguo Yu, Todd R Johnson, and Ramakanth Kavuluru. Phrase based topic modeling for semantic information processing in biomedicine. *Proc. Int. Conf. Mach. Learn. Appl.*, Vol. 2013, pp. 440–445, December 2013.

[8] Henry Widdowson. J.R. firth, 1957, papers in linguistics 1934–51. *Vigo Int. J. Appl. Linguist.*, Vol. 17, No. 3, pp. 402–413, November 2007.

[9] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. January 2013.

[10] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543. aclweb.org, 2014.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. October 2018.

[12] Dimo Angelov. Top2Vec: Distributed representations of topics. August 2020.

[13] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. May 2014.

[14] Daniel Cer, Yinfei Yang, Sheng-Yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Universal sentence encoder. March 2018.

[15] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-Networks. August 2019.

[16] Shufan Wang, Laure Thompson, and Mohit Iyyer. Phrase-BERT: Improved phrase embeddings from BERT with an application to corpus exploration. September 2021.

[17] David Blei and John Lafferty. Correlated topic models. *Adv.*

Neural Inf. Process. Syst., Vol. 18, p. 147, 2006.

[18] Lan Du, Wray Buntine, and Mark Johnson. Topic segmentation with a structured topic model. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 190–200. aclweb.org, 2013.

[19] Baoji Li, Wenhua Xu, Yuhui Tian, and Juan Chen. A phrase topic model for large-scale corpus. In *2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*, pp. 634–639, April 2019.

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. October 2013.

[21] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, Vol. 5, pp. 135–146, 2017.

[22] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pp. 2227–2237. cs.ubc.ca, 2018.

[23] Yaru Hao, Li Dong, Furu Wei, and Ke Xu. Visualizing and understanding the effectiveness of BERT. August 2019.

[24] Lang Yu and Allyson Ettinger. Assessing phrasal representation and composition in transformers. October 2020.

[25] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823. cvfoundation.org, 2015.

[26] Matthew Henderson, Rami Al-Rfou, B Strope, Yun-Hsuan Sung, László Lukács, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and R Kurzweil. Efficient natural language response suggestion for smart reply. *ArXiv*, 2017.

[27] Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[28] Adina Williams, Nikita Nangia, and Samuel Bowman. A Broad-Coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[29] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. September 2016.

[30] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pp. 55–60. aclweb.org, 2014.

[31] Kalpesh Krishna, John Wieting, and Mohit Iyyer. Reformulating unsupervised style transfer as paraphrase generation. October 2020.

[32] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. April 2019.

[33] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800GB dataset of diverse text for language modeling. December 2020.

[34] Leland McInnes, John Healy, and James Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. February 2018.

[35] Leland McInnes, John Healy, and Steve Astels. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, Vol. 2, No. 11, p. 205, March 2017.

[36] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. December 2014.

[37] Umap: Uniform manifold approximation and projection for dimension reduction. `https://umap-learn.readthedocs.io/en/latest/`. Accessed: 2022-01-10.

[38] The hdbscan clustering library. `https://hdbscan.readthedocs.io/en/latest/index.html`. Accessed: 2022-01-10.

[39] Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, Stroudsburg, PA, USA, 2017. Association for Computational Linguistics.

[40] P D Turney. Domain and function: A Dual-Space model of semantic relations and compositions. *J. Artif. Intell. Res.*, Vol. 44, pp. 533–585, July 2012.

[41] Shima Asaadi, Saif Mohammad, and Svetlana Kiritchenko. Big BiRD: A large, fine-grained, bigram relatedness dataset for examining semantic composition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 505–516. aclweb.org, 2019.

[42] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and Others. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, Vol. 12, pp. 2825–2830, 2011.

[43] David Mimno, Hanna Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the 2011 conference on empirical methods in natural language processing*, pp. 262–272. aclweb.org, 2011.

[44] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. July 2019.

[45] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. MPNet: Masked and permuted pre-training for language understanding. April 2020.