Transformer over Pre-trained Transformer for Neural Text Segmentation with Enhanced Topic Coherence

Kelvin Lo[1] **Yuan Jin**[1] **Weicong Tan**[1] **Ming Liu**[2] **Lan Du**[1*] **Wray Buntine**[1]
[1]Faculty of Information Technology, Monash University, Australia
{kelvin.lo,yuan.jin,charles.tan,lan.du,wray.buntine}@monash.edu
[2]School of Information Technology, Deakin University, Australia
m.liu@deakin.edu.au

Abstract

This paper proposes a transformer over transformer framework, called Transformer^2, to perform neural text segmentation. It consists of two components: bottom-level sentence encoders using pre-trained transformers, and an upper-level transformer-based segmentation model based on the sentence embeddings. The bottom-level component transfers the pre-trained knowledge learnt from large external corpora under both single and pair-wise supervised NLP tasks to model the sentence embeddings for the documents. Given the sentence embeddings, the upper-level transformer is trained to recover the segmentation boundaries as well as the topic labels of each sentence. Equipped with a multi-task loss and the pre-trained knowledge, Transformer^2 can better capture the semantic coherence within the same segments. Our experiments show that (1) Transformer^2 manages to surpass state-of-the-art text segmentation models in terms of a commonly-used semantic coherence measure; (2) in most cases, both single and pair-wise pre-trained knowledge contribute to the model performance; (3) bottom-level sentence encoders pre-trained on specific languages yield better performance than those pre-trained on specific domains.

1 Introduction

Text segmentation is an NLP task that aims to break text into topically coherent segments by identifying natural boundaries of changes of topics (Hearst, 1994; Moens and De Busser, 2001; Utiyama and Isahara, 2001). It is critical in the sense that many downstream tasks can benefit from the resulting structured text, including text summarization, keyword extraction and information retrieval.

Both supervised and unsupervised learning have been applied to text segmentation. With the lack of large-quantity labels on supervised training (Koshorek et al., 2018), unsupervised modeling based on clustering (Choi, 2000; Chen et al., 2009), Bayesian methods (Du et al., 2013, 2015; Malmasi et al., 2017) and graph methods (Glavaš et al., 2016; Malioutov and Barzilay, 2006) have been proposed. However, with the advancement of self-learning and transfer learning on deep neural networks, there are more recent supervised modeling approaches proposed that aim to predict labeled segment boundaries on smaller datasets. (Koshorek et al., 2018; Xing et al., 2020; Barrow et al., 2020; Glava and Somasundaran, 2020)

To the best of our knowledge, the most straightforward remedy to the above problems is knowledge transfer and distillation from pre-trained models. The rich pre-trained knowledge enables the training of a more general segmentation model on a small labeled dataset. In this paper, we propose a transformer over pre-trained transformer framework that allows different types of pre-trained information regarding sentences to be distilled to their classification for text segmentation. More specifically, the contributions of our paper are as follows:

- Our framework leverages pre-trained (and fixed) transformers at the bottom level to transfer (as sentence encoders) both *individual* and *pairwise* knowledge regarding sentences to train an upper-level transformer for segmentation.
- The upper-level transformer is trained with a multi-task loss with different targets, including the segment labels and the (section) topic labels.
- Our framework outperforms state-of-the-art segmentation models in terms of the P_k metric(Beeferman et al., 1999) across several real-world datasets in different domains and languages.
- A comprehensive ablation study shows that each component of our framework, in most cases, is essential by contributing to its segmentation performance.
- A thorough empirical study shows the impacts

*Corresponding author

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 3334–3340

of language-specific and domain-specific pre-trained transformers as the sentence encoders on the segmentation performance.

2 Related Work

In this section, we review the past literature on the text segmentation models. These models can further be categorized into being unsupervised and supervised.

2.1 Unsupervised Segmentation Models

Unsupervised segmentation models are developed based on some text similarity measures. C99 (Choi, 2000), TextTiling (Hearst, 1997) and TopicTiling (Riedl and Biemann, 2012) partitions texts with inter-sentence similarity matrices, lexical co-occurrence patterns and topic information from latent Dirichlet allocation (LDA) (Blei et al., 2003) respectively. Sophisticated Bayesian models were also proposed to capture the statistical characteristics of segment (topic) generation, including topic ordering regularities (Du et al., 2014), native language characteristics (Malmasi et al., 2017) and topic identities (Mota et al., 2019). On the other hand, GraphSeg (Glavaš et al., 2016) and Malioutov and Barzilay (2006) has formulated text segmentation as graph problems.

2.2 Supervised Segmentation Models

Earlier supervised segmentation models (Galley et al., 2003; Hsueh et al., 2006; Koshorek et al., 2018) rely on heuristics-based and heavily engineered segment coherence features to train traditional classifiers (e.g. decision trees (Hsueh et al., 2006)) that learn the relationships between the features and the segment labels.

In recent years, deep neural network based segmentation models have started to emerge. A common structure for them is a two-level hierarchical network, which consist of bottom-level sentence encoder and upper-level segment boundary classifier. Variants of LSTM (Hochreiter and Schmidhuber, 1997) and Bi-LSTM are vastly used in both lower-level and upper-level models from previous studies. However, the implementations of upper-level models are more diverse among them. Koshorek et al. (2018) and Wang et al. (2018) have used Bi-LSTM to predict segment boundary directly, while SECTOR (Arnold et al., 2019) predicts the topic of sentence and segment boundary sequentially with LSTM. S-LSTM (Barrow et al., 2020) further improves the performance by incorporating the ideas of previous models. On the other hand, Xing et al. (2020) have introduced an auxiliary pairwise sentence coherence loss. A similar architecture is also used by Lukasik et al. (2020).

The closest model to ours is proposed in (Glava and Somasundaran, 2020)[1] where transformers are used for both the levels of the architecture. They also developed a semantic coherence measure on distinguishing pairs of genuine and fake text snippets as an auxiliary loss alongside the segment classification loss. However, their model does not leverage the rich and diverse knowledge extracted from pre-training tasks (e.g. masked language modeling) to encode sentences at the bottom level. Addressing this limitation, our model leverages this pre-trained knowledge for dealing with a paucity of segment labels (e.g. in specialised domains).

3 Transformer^2 Architecture

Our proposed model adopts the popular two-level network architecture for text segmentation, which consists of a lower-level sentence encoder and an upper-level segment boundary classifier.

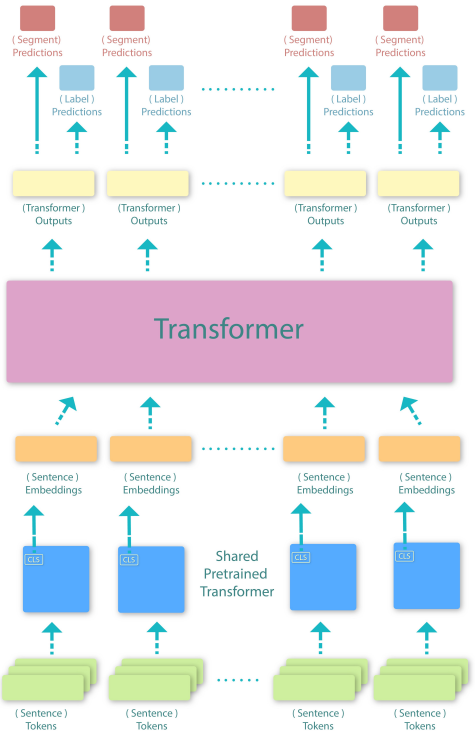
Our model aims to enhance the learning of semantic coherence between sentences from two aspects; 1) different pre-trained embeddings, generated from different NLP tasks on large external corpora, for the same sentences can capture rich and diverse information that the target corpus does not contain; 2) sentences within same segment(i.e. sharing same topic label) tend to be semantically more coherent than those across segments (i.e. with different topic labels). The above enhancements can further improve the segmentation performance of the transformer-based classifier.

3.1 Combining Different Pretrained Knowledge at the Bottom Level

To introduce different prior knowledge that describes different aspects (e.g. semantics, coherence, etc.) of each sentence into the segmentation, we combine different pre-trained sentence embeddings at the bottom level. More specifically, in this paper, we concatenate the embeddings respectively generated from the [CLS] tokens with single-sentence

[1]We have been unable to compare with their model as 1) their pre-trained model has not been made public and 2) rerunning their code incurs a major run-time error irrelevant to the dataset used and the data preprocessing procedures applied.

Figure 1: Transformer[2] Architecture



and pairwise-sentence inputs, that is the sentence embeddings $\boldsymbol{S} := [\boldsymbol{S}_{\text{single}}; \boldsymbol{S}_{\text{pairwise}}]$.

The single-sentence embeddings learned through masked language modelling (MLM) provide localised sentence information, while pairwise-sentence embeddings provide coherence information between consecutive sentences inherited from pairwise sentence classification tasks of pre-trained models, such as next-sentence prediction (NSP) (Devlin et al., 2019) and the sentence-order prediction (SOP) (Lan et al., 2019). Further details are summarised in table 1.

Table 1: Transformers leveraged for the bottom level in our experiments by combining their [CLS] embedding outputs pre-trained respectively under single and pairwise tasks.

	Single	Pairwise		
Transformer	**MLM**	**MLM**	**NSP**	**SOP**
BERT	✓	✓	✓	
XLNet	✓	✓		
RoBERTa	✓	✓		
ALBERT	✓	✓		✓

3.2 Sentence Classification at the Upper Level

Once the sentence embeddings are obtained, we train a transformer model at the upper level of the architecture to classify 1) whether each sentence is the segment boundary and 2) the topic label of each sentence. Thus, the loss function for the upper-level transformer can be formulated as follows:

$$\begin{aligned} L(\boldsymbol{y}_{\text{seg}}, \boldsymbol{y}_{\text{topic}}; \boldsymbol{S}, \boldsymbol{\Theta}) &= L_{\text{seg}}(\boldsymbol{y}_{\text{seg}}, \hat{\boldsymbol{y}}_{\text{seg}}; \boldsymbol{S}, \boldsymbol{\Theta}) \\ &\quad + L_{\text{topic}}(\boldsymbol{y}_{\text{topic}}, \hat{\boldsymbol{y}}_{\text{topic}}; \boldsymbol{S}, \boldsymbol{\Theta}) \\ \hat{\boldsymbol{y}}_{\text{seg}} &:= \text{Sigmoid}(\text{Linear}_2(\text{Transformer}_{\boldsymbol{\Theta}}(\boldsymbol{S}))) \\ \hat{\boldsymbol{y}}_{\text{topic}} &:= \text{Softmax}(\text{Linear}_K(\text{Transformer}_{\boldsymbol{\Theta}}(\boldsymbol{S}))) \end{aligned} \tag{1}$$

where $\boldsymbol{S} =< \boldsymbol{s}_1, \boldsymbol{s}_2, ..., \boldsymbol{s}_I >$, in this case, is the concatenation[2] of a sequence of embeddings of all the $\boldsymbol{I}$ sentences in the document[3]; $\boldsymbol{y}_{\text{seg}}, \boldsymbol{y}_{\text{topic}}$ are the binary segmentation and K topic labels for each sentence, while $\hat{\boldsymbol{y}}_{\text{seg}}, \hat{\boldsymbol{y}}_{\text{topic}}$ are their respective predictions. Correspondingly, linear layers with the respective output dimensions are put on top of the transformer with parameters $\boldsymbol{\Theta}$. The term L_{topic} denotes an auxiliary loss on the topic labels of each sentence. Minimizing this loss forces our framework to learn semantic coherence between sentences to account for their topical similarity. As for model training, the binary segmentation loss L_{seg} and the topic prediction loss L_{topic} are minimized respectively as the binary and categorical cross entropy losses with respect to $\boldsymbol{\Theta}$.

Table 2: Summary of WikiSection Dataset

Language	**Topic**	**Abbrev.**	**#Subtopics**	**#Documents**
English	Disease	en_disease	27	3,590
English	City	en_city	30	19,539
German	Disease	de_disease	25	2,323
German	City	de_city	27	12,537

4 Experimental Results

4.1 Datasets

We used the WikiSection dataset (Arnold et al., 2019) to evaluate the segmentation performance of our framework. It contains 38,000 full-text documents with segment information from English and German Wikipedia, each divided by topics regarding diseases and cities. The details of the corpora are summarised in Table 2.

4.2 Experimental Design

In the experiments, we leveraged both the single-sentence and pairwise-sentence pre-trained knowledge from the transformers specified in Table 1 to encode sentences at the bottom level. We aim to study the effects of bottom-level sentence encoders

[2]With a slight abuse of notation, we reuse the symbol $\boldsymbol{S}$ from Section 3.1 to denote a sequence of all the sentences in the document.

[3]$\boldsymbol{I}$ denotes the maximum number of sentences in a document including the paddings.

Table 3: Transformer models and their configurations (i.e. languages and domains) used in our experiments

Model	Config.	en_city	en_disease	de_city	de_disease
BERT	English	✓	✓		
	German			✓	✓
	BioClinical		✓		✓
XLNet	English	✓	✓		
RoBERTa	English	✓	✓		
	BioMed		✓		✓
ALBERT	English	✓	✓		

with different 1) transformer models, 2) languages and 3) domains on the segmentation performance.

Table 3 displays the details of the transformers and their configurations (i.e. languages and domains) used in the experiments. More specifically, we encoded the German corpora, i.e. **de_city** and **de_disease**, with German BERT, which is pre-trained on the German Wikipedia dump. Likewise, we also encoded the domain-specific corpora, i.e. **en_disease** and **de_disease**, with BioClinical models, pre-trained on the MIMIC III (Johnson et al., 2016) medical datasets. Detailed model configurations are listed in Appendix 4.4.

4.3 Evaluation Metrics & Baselines

Aligning with previous models, we evaluated the model performance with respect to the P_k metric proposed by Beeferman et al. (1999). It is a probabilistic metric that indicates, given a pair or words with k words apart, how likely will they lie in different segments. P_k values closer to 0 indicate the predicted segments are closer to ground truth, In our experiment, the value of k is set to be half of the average ground-truth segment length (Pevzner and Hearst, 2002).

The baselines include 1) machine learning segmentation models: C99 (Choi, 2000) and TopicTiling (Riedl and Biemann, 2012), and 2) state-of-the-art deep neural models: TextSeg (Koshorek et al., 2018), SECTOR (Arnold et al., 2019) with pre-trained embeddings, S-LSTM (Barrow et al., 2020) and BiLSTM+BERT (Xing et al., 2020). We followed the default hyper-parameter settings for all the models as specified in their official implementations.

4.4 Transformer^2 Settings

For all the corpora, we have fixed several hyper-parameters of Transformer^2. We have used the Adam optimiser (Kingma and Ba, 2015) with the learning rate being 0.0001. The maximum input sequence length was fixed at 150 sentences, as more than 94% of the documents have less than or equal to this number of sentences across the text segmentation corpora. Moreover, our model has 5 transformer encoder layers with 24 self-attention heads. Each of the encoder layers has a point-wise feed-forward layer of 1,024 dimensions. For the segmentation predictions, 70% of the inner sentences were randomly masked while all the begin sentences were not masked in order to address the imbalance class problem.

Table 4: P_k values of the baselines and the best variants of Transformer^2 for the different datasets; Bold and underscore figures indicate the best and second best results respectively.

Model	en_disease	de_disease	en_city	de_city
C99	37.4	42.7	36.8	38.3
TopicTiling	43.4	45.4	30.5	41.3
TextSeg	24.3	35.7	19.3	27.5
SECTOR+emb	26.3	27.5	15.5	16.2
S-LSTM	20.0	18.8	<u>9.1</u>	9.5
BiLSTM+BERT	21.1	28.0	9.3	11.3
Transformer$^2_{\text{XLNet}}$	25.2	-	11.7	-
Transformer$^2_{\text{ALBERT}}$	59.1	-	43.6	-
Transformer$^2_{\text{RoBERTa}}$	57.2	-	22.7	-
Transformer$^2_{\text{BERT}}$	**18.8**	-	<u>9.1</u>	-
without $\boldsymbol{S}_{\text{single}}$	<u>19.9</u>	-	**8.2**	-
Transformer$^2_{\text{de_BERT}}$	-	**16.0**	-	<u>7.3</u>
without $\boldsymbol{S}_{\text{single}}$	-	<u>17.1</u>	-	**6.8**

4.5 P_k results

Comparison with previous models[4] Table 4 shows the performance of the best variants of Transformer^2 for different datasets and that of the baseline models in terms of the P_k metric. Our models Transformer$^2_{\text{BERT}}$ and Transformer$^2_{\text{de_BERT}}$ outperforms all previous models by a notable margin in English and German corpus respectively.

Ablation study of model components We have examined the effects of single and pairwise embeddings, joint modeling on topic classification and choice of lower-level sentence encoder, summarised in tables 5 and 6. The results from table 5 shows the models yield better results without the single sentence embeddings $\boldsymbol{S}_{\text{single}}$ on the en_city and de_city datasets. This suggests that combining different pre-trained knowledge does not always improve the segmentation quality.

The results also show that the segmentation quality solely based on the change in topic label prediction labels is significantly inferior than using the segmentation labels. This is because predicting the same topic label consecutively in a multi-class setting is more difficult than the same segment label consecutively in a binary-class setting.

[4]Detailed qualitative analysis can be found in Appendix 4.6

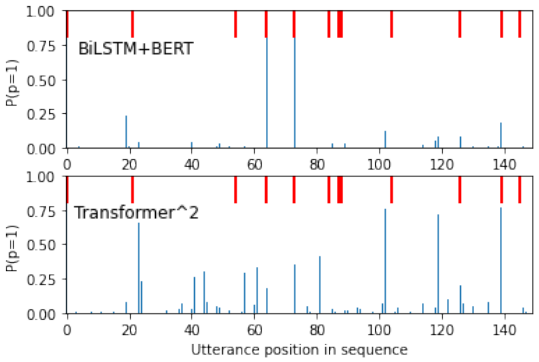
Table 5: An ablation study on the impacts of each component of the best variants of Transformer^2 on P_k

model	en_disease	de_disease	en_city	de_city
Transformer$^2_{\text{BERT}}$	**18.8**	-	9.1	-
without $\boldsymbol{S}_{\text{single}}$	19.9	-	**8.2**	-
without $\boldsymbol{S}_{\text{pairwise}}$	19.2	-	9.1	-
without L_{topic}	20.4	-	**8.2**	-
without L_{seg}	25.3	-	41.1	-
Transformer$^2_{\text{de_BERT}}$	-	**16.0**	-	7.3
without $\boldsymbol{S}_{\text{single}}$	-	17.1	-	**6.8**
without $\boldsymbol{S}_{\text{pairwise}}$	-	18.8	-	9.2
without L_{topic}	-	19.5	-	7.2
without L_{seg}	-	20.2	-	27.5

Table 6: P_k values of the domain-specific BERT and RoBERTa

Sentence Encoder	en_disease	de_disease
Transformer$^2_{\text{BioClinical_BERT}}$	21.4	45.8
Transformer$^2_{\text{BioMed_RoBERTa}}$	36.4	50.2

Figure 2: Probabilities of segment boundaries compared to the gold-standard ones (red lines on top of each graph) on one en_disease document where Transformer^2's predicted probabilities are more dispersed and accurate.



On the other hand, from table 6, we can observe that models pre-trained on corpora in specific domains, such as BioClinical BERT, do not improve text segmentation quality compared to models pre-trained on giant language-specific corpora, such as German BERT, which is accountable to the tokenization quality of such model.

4.6 Qualitative Analysis of Transformer^2

Apart from the quantitative evaluation based on the P_k metric, we also conducted qualitative analysis on the segment predictions from both our model and the most competitive baseline: BiLSTM+BERT. More specifically, we randomly picked up several documents from en_disease and de_disease datasets, visually inspected and then summarised the difference between the segmentation styles of the best variants of Transformer^2 and BiLSTM+BERT. We find that the variants of Transformer^2 tend to yield **more dispersed** segment predictions across the documents, while the predictions of BiLSTM+BERT tend to be **more concentrated** and often documents are clustered as one big segment. Figure 2 shows one such example of our finding.

5 Conclusion and Future Work

In this paper, we propose a transformer over pre-trained transformer framework, called Transformer^2, for text segmentation with a focus on enhancing the learning of the semantic coherence between sentences. The bottom level of Transformer^2 combines (untrainable and fixed) sentence embeddings outputted respectively from transformers pre-trained with both the single-sentence and the pairwise-sentence NLP tasks. An upper-level transformer is trained upon the combined sentence embeddings to minimize both the binary segmentation loss and the auxiliary topic prediction loss.

The empirical results show that the best variants of Transformer^2 outperform several state-of-the-art segmentation models, including the deep neural models, across four real-world datasets in terms of a commonly-used segment coherence measure P_k. We have also conducted a comprehensive ablation study which shows that in most cases, each component of Transformer^2 is helpful for boosting the segmentation performance. We have also found that using language-specific pre-trained transformers at the bottom level is more useful than using domain-specific ones. For the future work, we will investigate the efficacy of Transformer^2 on helping the downstream NLP tasks such as text summarisation, keyword extraction and topic modelling.

References

Sebastian Arnold, Rudolf Schneider, Philippe Cudré-Mauroux, Felix A. Gers, and Alexander Löser. 2019. Sector: A neural model for coherent topic segmentation and classification. *Transactions of the Association for Computational Linguistics*, 7:169–184.

Joe Barrow, Rajiv Jain, Vlad Morariu, Varun Manjunatha, Douglas Oard, and Philip Resnik. 2020. A joint model for document segmentation and segment labeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 313–322.

Doug Beeferman, Adam Berger, and John Lafferty.

1999. Statistical models for text segmentation. *Machine learning*, 34(1-3):177–210.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Harr Chen, S.R.K. Branavan, Regina Barzilay, and David R. Karger. 2009. Global models of document structure using latent permutations. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 371–379.

Freddy Y. Y. Choi. 2000. Advances in domain independent linear text segmentation. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. 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)*, pages 4171–4186.

Lan Du, Wray Buntine, and Mark Johnson. 2013. 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*, pages 190–200.

Lan Du, John K Pate, and Mark Johnson. 2014. Topic models with topic ordering regularities for topic segmentation. In *2014 IEEE International Conference on Data Mining*, pages 803–808. IEEE.

Lan Du, John K Pate, and Mark Johnson. 2015. Topic segmentation with an ordering-based topic model. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, pages 2232–2238.

Michel Galley, Kathleen McKeown, Eric Fosler-Lussier, and Hongyan Jing. 2003. Discourse segmentation of multi-party conversation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 562–569.

Goran Glava and Swapna Somasundaran. 2020. Two-level transformer and auxiliary coherence modeling for improved text segmentation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:7797–7804.

Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. 2016. Unsupervised text segmentation using semantic relatedness graphs. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 125–130.

Marti A. Hearst. 1994. Multi-paragraph segmentation expository text. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 9–16.

Marti A. Hearst. 1997. Text tiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Pei-Yun Hsueh, Johanna D Moore, and Steve Renals. 2006. Automatic segmentation of multiparty dialogue. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3:160035.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Omri Koshorek, Adir Cohen, Noam Mor, Michael Rotman, and Jonathan Berant. 2018. Text segmentation as a supervised learning task. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 469–473.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Michal Lukasik, Boris Dadachev, Kishore Papineni, and Gonçalo Simões. 2020. Text segmentation by cross segment attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4707–4716.

Igor Malioutov and Regina Barzilay. 2006. Minimum cut model for spoken lecture segmentation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 25–32.

Shervin Malmasi, Mark Dras, Mark Johnson, Lan Du, and Magdalena Wolska. 2017. Unsupervised text segmentation based on native language characteristics. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1457–1469.

Marie-Francine Moens and Rik De Busser. 2001. Generic topic segmentation of document texts. In *Proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval*, pages 418–419.

Pedro Mota, Maxine Eskenazi, and Luísa Coheur. 2019. BeamSeg: A joint model for multi-document segmentation and topic identification. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 582–592.

Lev Pevzner and Marti A. Hearst. 2002. A critique and improvement of an evaluation metric for text segmentation. *Comput. Linguist.*, 28(1):19–36.

Martin Riedl and Chris Biemann. 2012. TopicTiling: A text segmentation algorithm based on LDA. In *Proceedings of ACL 2012 Student Research Workshop*, pages 37–42.

Masao Utiyama and Hitoshi Isahara. 2001. A statistical model for domain-independent text segmentation. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, pages 499–506.

Yizhong Wang, Sujian Li, and Jingfeng Yang. 2018. Toward fast and accurate neural discourse segmentation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 962–967.

Linzi Xing, Brad Hackinen, Giuseppe Carenini, and Francesco Trebbi. 2020. Improving context modeling in neural topic segmentation. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 626–636.