Breaking the Narrative: Scene Segmentation through Sequential Sentence Classification

Murathan Kurfalı
Department of Linguistics
Stockholm University
Stockholm, Sweden
murathan.kurfali@ling.su.se

Mats Wirén
Department of Linguistics
Stockholm University
Stockholm, Sweden
mats.wiren@ling.su.se

Abstract

In this paper, we describe our submission to the Shared Task on Scene Segmentation (STSS). The shared task requires participants to segment novels into coherent segments, called scenes. We approach this as a sequential sentence classification task and offer a BERT-based solution with a weighted cross-entropy loss. According to the results, the proposed approach performs relatively well on the task as our model ranks first and second, in official in-domain and out-domain evaluations, respectively. However, the overall low performances (0.37 F_1-score) suggest that there is still much room for improvement.

1 Introduction

Scene segmentation is a novel task introduced in (Zehe et al., 2021a) that aims to divide long narrative texts, e.g. novels, into smaller coherent segments or scenes, as they are called. Scenes, in this context, can be roughly defined as "a segment of a text where the story time and the discourse time are more or less equal, the narration focuses on one action and space and character constellations stay the same" (Zehe et al., 2021a).[1] The task of scene segmentation is of great value on several ends: (i) it can be directly employed in several digital humanities tasks, e.g. plot reconstruction; (ii) segmenting longer texts into smaller coherent pieces help other NLP tasks, e.g. co-reference resolution, that struggle with texts longer than a couple of paragraphs (Joshi et al., 2020); (iii) as a novel task that requires high-level modeling of long texts, it offers itself as a valuable probing task to evaluate language models on long-context scenarios which is an active research area (Tay et al., 2020).

Our main interest in the current paper is to explore whether scene segmentation can be handled as a sequential sentence classification task. To this end, we follow the methodology proposed in Cohan et al. (2019), which encodes all sentences in a sequence jointly through BERT (Devlin et al., 2019) to directly leverage the contextual information from all tokens in the sequence at the same time. The model of Cohan et al. (2019) is further adapted to the task via introduction of a weighted cross-entropy loss in order to account for the imbalanced distribution of the labels in the dataset.

According to the official results, our model achieves the best performance on the in-domain texts, significantly outperforming the second-ranking system. However, the performance drops when evaluated on out-of-domain novels, suggesting that the proposed methodology only poorly generalizes over different domains. We release our system to facilitate reproducibility and future work.[2]

2 System Overview

2.1 Task Details

The scene segmentation task can be framed in several ways. Within the shared task, it is defined as the identification of the boundaries that delimit the consecutive segments (Zehe et al., 2021b). The boundaries between segments are labeled according to the types of segments they delimit. Specifically, a boundary can belong one of the following three classes: *Scene-Scene*; *Nonscene-Scene*; *Scene-Nonscene*.[3]

The participating teams are evaluated only according to their success at finding and labeling

[1] Interested readers are referred to annotation guidelines available at https://zenodo.org/record/4457177 for further details.

[2] https://github.com/MurathanKurfali/scene_segmentation

[3] Unlike Scenes, Nonscenes, naturally, are not distinguished from one another; hence, Nonscene-Nonscene is not a valid transition.

these boundaries. That is to say, classification of an individual sentence as belonging to a Scene or a Nonscene means very little in the evaluations. Of the possible three transitions, *Scene-Scene* is the most common one as Nonscenes are significantly less frequent in data (see Table 1).

2.2 Our Model

We model scene segmentation as a Sequence Sentence Classification (SSC) task where the goal is to understand whether a given sentence is segment-initial or not along with the type of segment it belongs to. Similarly to the more common token classification tasks, e.g. POS-tagging or NER, we employ the IOB2 format and assign a tag to each sentence. Specifically, we label segment-initial sentences (boundaries) as *#X-B* and other sentences as merely *#X* where *X* indicates the type of segment (Scene or Nonscene).

Our classifier closely follows the methodology proposed in Cohan et al. (2019). Here, the authors employed BERT to perform several document-level classification tasks, e.g. *abstract sentence classification*, where the aim is to classify sentences in a scientific abstract into their rhetorical roles such as introduction, method, etc. The rest of this section describes the model along with our modifications.

The proposed methodology follows the standard way of using BERT through fine-tuning on the target task but uses a novel input representation. The classifier used in the experiments is illustrated in Figure 1. As input, a sequence of N sentences is concatenated by BERT's special delimiter token `[SEP]`, yielding one long sequence. This sequence, after the insertion of the standard `[CLS]` token at the beginning, is fed into BERT. However, unlike the standard way of using the `[CLS]` token as the representation of the input sequence, the representations of the individual `[SEP]` tokens are used as the representations of the sentences that precede them. Hence, instead of the `[CLS]` token, `[SEP]` representations are classified by a multi-layer feedforward network to reach labels.

The rationale for using `[SEP]` as sentence representation has to do with the next-sentence objective of BERT: "Intuitively, through BERT's pretraining, the `[SEP]` tokens learn sentence structure and relations between continuous sentences" (Cohan et al., 2019). During fine-tuning, the model is further primed to assign appropriate weights to `[SEP]` tokens to encode necessary contextual information

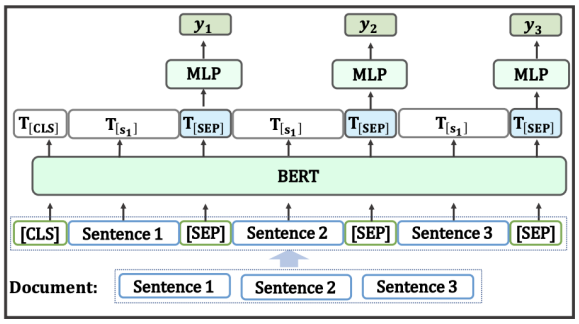

Figure 1: Overview of the system architecture. Each sentence is represented by the respective `[SEP]` token which is used to predict the label. Figure copied from Cohan et al. (2019).

for classification. Fine-tuning BERT in this way has the benefit of simultaneously leveraging the contextual information from all sentences in the sequence.

Loss function The model is trained to minimize the cross-entropy loss between the probabilities over the possible labels computed using a softmax activation and the target distribution. However, during the initial experiments, we observed that the model severely suffered from the highly skewed label distribution, namely the low number of boundary sentences in comparison to non-boundary ones.[4] In order to mitigate this issue, following the previous studies (Rotsztejn et al., 2018; Cui et al., 2019; Yang et al., 2019), we introduce a weighting factor to the loss function where each class is assigned a weight that is inversely proportionally to their frequency in the training set:

$$weight_c = \frac{\sum_i freq(i)}{freq(c)}$$

where *freq* indicates the count of a certain class. Overall, the weighted cross-entropy becomes $Loss_{WCE} = -\sum_c^C w_c t_c log(s_c)$ where w_c is the weight, t_c is the gold truth value (taking either 0 or 1), and s_c is the corresponding Softmax probability of the class c.

3 Experimental Setup

3.1 Data

The dataset used in this shared task is based on an expanded version of the annotation effort introduced in (Zehe et al., 2021a), and consists of 20 German novels in total, excluding the blind test sets

[4]The most frequent label, *Scene*, single-handedly accounts for 96.1% of the training data.

Split	Scene			Non-scene		
	Count	$\|Avg.\ Segment\|$	$\|Avg.\ Sent\|$	Count	$\|Avg.\ Segment\|$	$\|Avg.\ Sent\|$
Training	1075	45.33	10.58	51	16.11	15.39
Dev	127	69.5	12.12	7	5.6	18.20
Test	46	38.16	8.05	7	19.14	11.15

Table 1: Characteristics of the train/dev/test splits used in model development. The numbers in the columns refer to number of segments, the average size of a segment (in terms of # of sentences) and the average size of a sentence (in terms of # of words) for Scenes and Non-scenes separately.

used in the official evaluations. During model development, we create custom development and test sets by randomly allocating one file for each, using the remaining 18 files for training.[5] The statistics regarding the training/dev/test splits used during model development are provided in Table 1.

3.2 Parameter Setting

We follow the implementation of Cohan et al. (2019).[6] As the language model, we use the large German BERT model from (Chan et al., 2020) (dubbed GBERT-large[7]) due its superior performance over the existing German models. The batch size of 8 and gradient accumulation steps of 4 are used to reach effective batch size of 32. All experiments are run on a single V100 GPU. We set the learning rate to 5e-6 and the training is run for the maximum of 100 epochs with the early stopping applied (patience = 20) based on the performance on the development set. Due to BERT's inherit sequence size limit, we set a threshold of 25 sentences in each sequence which is chosen empirically (i.e., according to the performance on the in-house test set) among the set of $\{20, 25, 30, 50\}$.

4 Results and Discussion

The official evaluation is performed on two different test sets:

i. Test suite 1 focuses on in-domain evaluation and consists of 5 annotated dime novels,

ii. Test suite 2 focuses on out-of-domain evaluation and consists of 2 annotated contemporary high-literature texts.

Table 2 presents the breakdown of our results into each possible transition whereas the official ranking of the participating systems, according to the mean micro-averaged F_1 scores, is provided in Table 3. According to the official rankings, the proposed approach is good at segmenting in-domain novels and outperforms the second best system by some margin. However, the performance significantly drops when evaluated on out-of-domain novels, suggesting that the system generalizes poorly across domains.

According to Table 2, our model is best at recognizing Scene to Scene transitions; however, it is almost completely incapable of finding the borders between non-scenes and scenes. Suggested by the high-recall, low-precision scores, our model tends to over-segment the novels. On average, the system divides the in-domain novels into 1.76 and out-of-domain novels into 1.61 times more segments. This tendency towards over-segmenting hints at over-sensitivity to certain markers which is further discussed in the next section.

Overall, the results clearly demonstrate that the task is extremely challenging even in the in-domain setting. The poor performance of solutions based on contextual embeddings[8] highlight the need for novel architectures. One obvious drawback with BERT-based models is their inability to encode long sequences. Hence, a straightforward extension of the current model would be to employ a model which supports longer contexts, e.g. Longformer (Beltagy et al., 2020); however, such a model is unfortunately not available for German at the time of writing.

5 Error Analysis

In addition to the official evaluation, we performed a manual error analysis of our model's predictions on the in-house test set (see Section 3.1). One observation was that in certain cases, although the model correctly recognized the type of transition (e.g. Scene-Scene), it misplaced the boundary only

[5] Files *9783740941093* and *9783732522033* are used the dev and test set, respectively.

[6] https://github.com/allenai/sequential_sentence_classification

[7] https://huggingface.co/deepset/gbert-large

[8] A BERT-based baseline in the original resource paper similarly fails on this task (Zehe et al., 2021a).

	In-domain			Out-of-domain		
	Prec.	Rec.	F_1-score	Prec.	Rec.	F_1-score
Scene-to-Scene	0.31	0.64	0.42	0.14	0.26	0.19
Scene-to-Nonscene	0.08	0.06	0.07	0.00	0.00	0.00
Nonscene-to-Scene	0.00	0.00	0.00	0.00	0.00	0.00
Micro average	0.29	0.51	0.37	0.14	0.22	0.17
Macro average	0.13	0.23	0.16	0.05	0.09	0.06
Weighted average	0.25	0.51	0.33	0.12	0.22	0.16

Table 2: Official results of our submission (prec(ision, rec(all) and F_1-score) for each type of transition along with the averaged results

Rank	Track 1	Track 2
1.	**0.37**	0.26
2.	0.16	**0.17**
3.	0.07	0.12
4.	0.02	0.11
5.	-	0.04

Table 3: Official rankings and results of all participating systems, according to the micro-averaged F_1 scores, averaged over all the novels in the corresponding suite. Results of our submission is highlighted in boldface.

by a single sentence. An instance of this can be seen in Example (1), where the predicted boundary appears just before sentence (1a), whereas the gold boundary appears just before the subsequent sentence (1b):[9]

(1) a. Und bald darauf fuhr der Wagen aus dem Wald und einen allmählich ansteigenden Berg hinan.
(*And soon afterwards the car drove out of the forest and up a gradually rising mountain.*)

b. Dort oben lag das Schloss Treuenfels.
(*Treuenfels Castle was up there.*)

Furthermore, as mentioned in the previous section, the system tends to over-segment the novels. A manual inspection of false positives (sentences that are erroneously identified as segment boundaries) reveals that despite being incorrect, these predictions are not completely random. Most of the false positives involve an adverbial or other kind of phrase which signals a shift in time and/or place. Some cherry-picked examples are given in Examples 2–5:[10]

(2) Als das Gefährt das Bergplateau erreicht hatte, ließ der Fahrer einige Male laut die Hupe ertönen.
(*When the vehicle had reached the mountain plateau, the driver sounded the horn a few times.*)

(3) Eines Abends, als Graf Harro von einer Herrengesellschaft zeitiger nach Hause kam, als man erwartete, fand er seine Gattin in einer sehr zärtlichen Stellung mit dem jungen Prinzen.
(*One evening, when Count Harro came home earlier than expected from a gentlemen's company, he found his wife in a very affectionate position with the young prince.*)

(4) Und am nächsten Morgen fand man die Gräfin Alice tot auf ihrem Lager.
(*And the next morning the Countess Alice was found dead in her bed.*)

(5) Er wandte sich um und ging wieder zurück, bis in das Zimmer, wo der Schreibtisch der Gräfin Alice stand.
(*He turned and went back to the room where Countess Alice's desk was.*).

Similar to the behavior of the baseline system proposed in Zehe et al. (2021a), these examples highlight the model's sensitivity to the local cues rather than the larger context. That is, to a certain extent, the system makes its predictions according to the individual phrases that signal shifts in time or place, paying too little attention to the global context.

[9]The English translations have been produced by Google Translate.

[10]In these examples, the system has predicted the shift immediately before the sentences displayed.

6 Conclusion

The current paper summarizes our submission to the Shared Task on Scene Segmentation (STSS). We handle scene segmentation as a sequential sentence classification task and offer a BERT-based solution. The proposed model achieves the best performance in the in-domain evaluations but falls short of transferring its performance across domains. Error analysis further reveals that the predictions are more sensitive to local cues rather than the global structure of the text, highlighting the need for better document-level modeling.

Acknowledgments

This work has been partly funded by an infrastructure grant from the Swedish Research Council (SWE-CLARIN, 2019–24; contract no. 2017-00626). We thank the Swedish National Infrastructure for Computing (SNIC) for providing computational resources under Project 2020/33-26.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796.

Arman Cohan, Iz Beltagy, Daniel King, Bhavana Dalvi, and Dan Weld. 2019. Pretrained language models for sequential sentence classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3693–3699, Hong Kong, China. Association for Computational Linguistics.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Jonathan Rotsztejn, Nora Hollenstein, and Ce Zhang. 2018. Eth-ds3lab at semeval-2018 task 7: Effectively combining recurrent and convolutional neural networks for relation classification and extraction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 689–696.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. 2020. Long range arena: A benchmark for efficient transformers. *arXiv preprint arXiv:2011.04006*.

Kisu Yang, Dongyub Lee, Taesun Whang, Seolhwa Lee, and Heuiseok Lim. 2019. Emotionx-ku: Bert-max based contextual emotion classifier. *arXiv preprint arXiv:1906.11565*.

Albin Zehe, Leonard Konle, Lea Katharina Dümpelmann, Evelyn Gius, Andreas Hotho, Fotis Jannidis, Lucas Kaufmann, Markus Krug, Frank Puppe, Nils Reiter, et al. 2021a. Detecting scenes in fiction: A new segmentation task. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3167–3177.

Albin Zehe, Leonard Konle, Svenja Guhr, Lea Katharina Dümpelmann, Evelyn Gius, Andreas Hotho, Fotis Jannidis, Lucas Kaufmann, Markus Krug, Frank Puppe, Nils Reiter, and Annekea Schreiber. 2021b. Shared task on scene segmentation@konvens2021. In *Shared Task on Scene Segmentation*.