LTUHH@STSS: Applying Coreference to Literary Scene Segmentation

Hans Ole Hatzel
Language Technology Group
Universität Hamburg, Germany
hatzel@informatik.uni-hamburg.de

Chris Biemann
Language Technology Group
Universität Hamburg, Germany
biemann@informatik.uni-hamburg.de

Abstract

In this work, we describe a system for scene segmentation that, relying on character constellations as one of the defining characteristics of scenes, employs a state-of-the-art coreference system. Conceptually building on one of the presented baseline systems, we use a transformer model, enhanced with additional coreference-based features, to identify scene boundaries on the basis of sentence pairs. Finding one of our system's core weaknesses to lie in its local decision making, we adapt an equidistance constraint, avoiding the common error of predicting very short scenes that in many cases only cover a single sentence. We show that coreference is a suitable feature for scene segmentation and experiment with dynamic programming approaches for non-local decisions. This work is a submission for the shared task scene segmentation (STSS) held at KONVENS 2021, where task participants were asked to, given annotated training data, build systems that split novels into scenes: segments narrating a coherent action in one location with the same characters. Our system ranks 4/4 and 4/5 in Track 1 and Track 2, respectively.

1 Introduction

One of the most defining characteristics of scenes are character constellations, in this work we describe a scene segmentation system exploiting this characteristic. Other defining aspects of scenes such as the story and discourse time being equal and the fact that they contain a coherent sequence of actions will not be explicitly modeled in this work. The shared task scene segmentation hosted by Zehe et al. (2021b) provides training data in the form of 22 dime novels, with an additional (for the task duration) unpublished test set and a single trial document. We chose a transformer-based approach as a starting point; we use BERT (Devlin et al., 2019) for scene segmentation, following the general approach of the best baseline proposed by (Zehe et al., 2021a). Further, we enrich the BERT-based representation using two sets of features, **(a)** a coreference-based approach to finding the characters in a given scene and **(b)** a set of surface features we believe may be helpful. In a second step, we improve our model's results by adding non-local decisions in the form of a cost function optimized using a dynamic programming technique.

2 Related Work

Pethe et al. (2020) approach the task of chapter segmentation, the task of splitting a document into its chapters. This task is related to scene segmentation in that it operates on a similar domain. As we conjecture, chapter boundaries may also correspond with changes in location or characters, making this work more relevant still. Pethe et al. (2020) take an equidistant approach to chapter segmentation, thereby enhancing local decisions with the knowledge that chapter boundaries tend to be somewhat evenly placed throughout a novel. The equidistant approach is applied by minimizing the following equation:

$$cost(n,k)=\min_{i\in[0,n-1]}\left(cost(i,k-1)+(1-\alpha)\frac{|n-i|}{L}\right)-\alpha\cdot s_n$$

Where k is the number of breaks to be inserted, n the position at which to insert a break and L the target length of each segment. α is a hyperparameter controlling the impact of the local boundary score s_n with values approaching one placing more importance on local decisions.

In our previous work (Schröder et al., 2021), we trained state-of-the-art models for coreference resolution on German data. Following the coarse-to-fine inference architecture for coreference (Lee et al., 2018), we fine-tune transformer models on the German TüBa-D/Z dataset, adapting them to the literature domain using further fine-tuning on

the DROC dataset (Krug et al., 2018). While some of our models enable the handling of arbitrary length texts, in this work we only rely on the coarse-to-fine model the application of which, due to its memory requirement characteristics, is limited to shorter documents.

3 Model and Features

In order to maximize the contextual information input to BERT, we do not pass an explicit context in conjunction with the two sentences in question (unlike the baseline approach in Zehe et al., 2021a). Instead, our approach follows the Next Sentence Prediction (NSP) training objective in BERT. For each sentence boundary present in the input data, we predict if the sentence to either side is part of the same scene or if there is a boundary between them (i.e. we perform a binary classification for the input "`[CLS]` *scene_candidate_a* `[SEP]` *scene_canidate_b* `[SEP]`"). Note that in the context of the NSP task, "sentence" actually refers to any input sequence and not a sentence in the linguistic sense. We see this alignment with the NSP as a benefit of our system, enabling us to leverage more of BERT's pre-trained capabilities. For this reason, we also chose to use a BERT model rather than an Electra model (Clark et al., 2020), as Electra models are not trained on the NSP objective.

While we did experiment with a BERT model trained on German literary data[1], we did not find success with it which, we attributed to the fact that it is fine-tuned on named entity recognition and may have, in a case of catastrophic forgetting, lost the ability to perform the NSP task. While the coreference-based features rely on previous work of ours (Schröder et al., 2021), for all of the remaining feature extraction we used the "de_core_news_lg" model in spaCy (Honnibal et al., 2020). All features are passed into a linear layer with GELU activation function (Hendrycks and Gimpel, 2020) in conjunction with the pooled BERT output (i.e. the `[CLS]` token's embedding). Final predictions are made using individual linear layers for each of the three outputs: binary scene type labels for each of the two sequences and the binary decision of whether there is a scene boundary between them, each with sigmoid activation functions. The model is trained using SGD and binary-cross-entropy loss for each of the three labels, using class weighting based on the training data distribution.

[1] https://huggingface.co/severinsimmler/literary-german-bert

3.1 Coreference Features

Leveraging coreference features we seek to model one of the central components of scenes: the character constellations. To this end, we pass the number of unique characters appearing in each of the input sequences, together with the number of unique characters appearing in both sequences to the model.

Taking a more global approach to coreference would also be possible, in this case, the number of characters involved in the current context may be compared to the global number of characters. While this approach may yield further improvements, we did not test it, partly due to the fact that global coreference resolution for long documents still is much more susceptible to errors than local approaches (Schröder et al., 2021).

3.2 Named Entity Recognition Features

One feature that we, following manual inspection of the training data, expect to be predictive of scene boundaries are named entities. The explicit mention of characters as well as that of locations should indicate a scene change. We extract the named entity tags for persons, locations, and miscellaneous entities and use document-length-normalized counts of each of them as a model input. While the coreference features capture some similar information, they capture neither location mentions nor are they able to differentiate between explicit and anaphoric character mentions.

Using a NER system trained specifically on literary data could help this step, such data is available in the DROC dataset (Krug et al., 2018).

3.3 Surface Features

In an effort to improve our model, we added a set of surface features that we believed may be indicative of scene changes. We passed the number of tokens (including special characters such as quotes and punctuation) fulfilling different properties to our model

- being punctuation
- being uppercased
- being quotation marks
- being a stop word
- being the start of a sentence

While all these features could, in principle, be picked up by means of representation learning in our neural model, we still add them due to the relatively small number of training samples.

4 Intermediate Results

While, in principle, our model is capable of predicting both scene boundaries and scene types, our final system uses two distinct models with the same architecture and inputs for the two tasks. Joint training presents non-trivial challenges in balancing the two target objectives but may yield improvements in final results. Both models were trained with early stopping on the trial data (i.e. one document provided with the task description but not as part of the training data); a hyperparameter search for individual learning rates for the final layers (between 1×10^{-3} and 1×10^{-5}) and the BERT model (between 1×10^{-4} and 2×10^{-5}) was performed using the Tree-structured Parzen Estimator (Bergstra et al., 2011) implementation by Akiba et al. (2019). The final model for scene types stopped after 5000 (returning to the set of weights from step 2000) steps of batch size 24 (with an evaluation frequency of 1000 steps) and used a learning rate of 9.9×10^{-5} for BERT and 6.4×10^{-4} for the final layers. The final model for scene types stopped after 18 000 (returning to the set of weights from step 15 000) steps of batch size 24 (with an evaluation frequency of 1000 steps) and used a learning rate of 4.8×10^{-5} for BERT and 2.84×10^{-5} for the final layers.

Using the features described so far we reach an F1-score of 33.7 on the task's trial document[2], presumably already outperforming the baseline system. Figure 1 illustrates the predicted boundaries together with the networks output values for each of the potential scene splits, i.e. each pair of sentences. Notably, there are multiple cases of two or more directly adjacent instances of false positives. Sometimes, like at the very end of the document, in conjunction with a true positive boundary. This illustrates what we see as a key weakness of our initial model; since decisions are purely local, when in doubt about the placement, the model creates multiple boundaries where one would be sufficient.

[2]Unless otherwise specified F1-score refers to the boundary class's F1-score throughout this document

5 Non-Local Model

As discussed in Section 4 we see an issue in the local nature of scene segmentation boundaries. One approach to remedy this may be, training on sequences of adjacent sentence pairs; this would have the advantage of allowing for non-local decisions, informed by any part of neighboring inputs. At the same time, however, this increases the memory requirements, and with scene boundaries occurring about every 43 sentences on average, a large enough context may (depending on available GPU memory) be infeasible to jointly train. Our early approaches instead focused on using neural sequence models on local decision outputs but using this approach we did not manage to improve upon local-decision-based results.

Instead, we chose a purely algorithmic approach without training: the dynamic programming (DP) approach by Pethe et al. (2020), a technique that requires prior knowledge of the number of chapters, or in our case scene, boundaries. Applying their approach to the task's trial document which was held-out, given the correct number of scene boundaries, (with $\alpha = 0.9$) results in an F1-score of 39.1. This represents is an improvement of around 5.4 on the local F1-score of 33.7. For comparison, when only using the k highest confidence values, where k is the number of gold boundaries, we only get an F1-Score of 34.8, illustrating that the mere knowledge of the number of scenes is not as impactful. Figure 2 shows the effect the cost function can have on decisions, while $\alpha = 0.7$ actually entails a worse F1-Score, the effect is very subtle when using larger α values (i.e. when incorporating local decisions to a larger extent).

Figure 3 illustrates that the coefficient of variation (CV) for the shared task's scene boundary is much higher than it is for the chapter data in the work by Pethe et al. (2020), where the distribution is centered around a value below 0.5. This can be interpreted as the length of chapters inside most documents being less variable than the length of scenes in many documents in our dataset. Although it is to be noted that the two statistics are made on the basis of very different datasets. The standard deviation of the distribution of average per document scene lengths (in sentences) is 10.84 with a mean of 45.3 and, accordingly, a CV of 0.24.

Another very simple approach to using non-local information is to, in a fixed window, only consider the top value to actually constitute a boundary. For

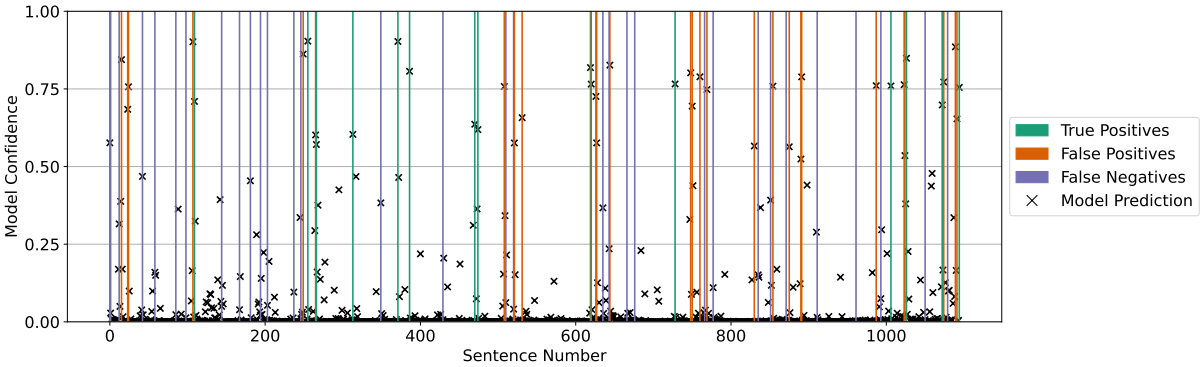

Figure 1: Positions of scene splits in the trial data using only local decisions

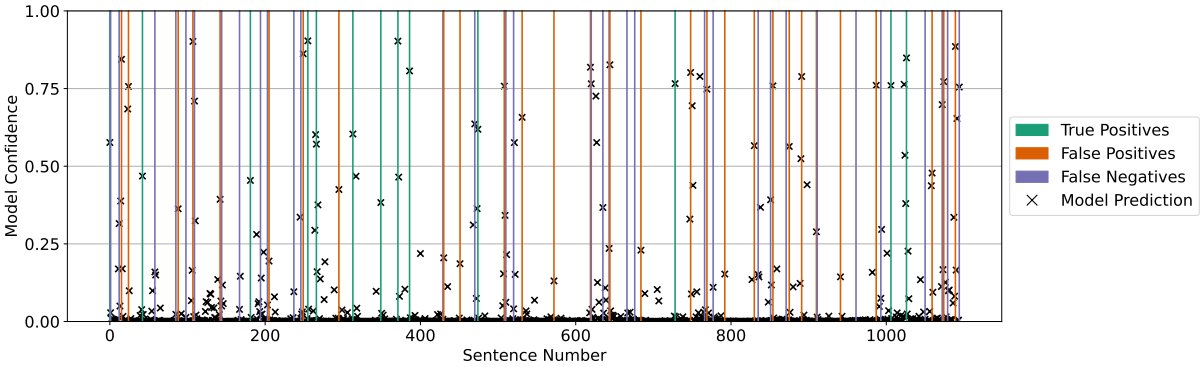

Figure 2: Positions of scene splits using the DP technique with $\alpha = 0.7$

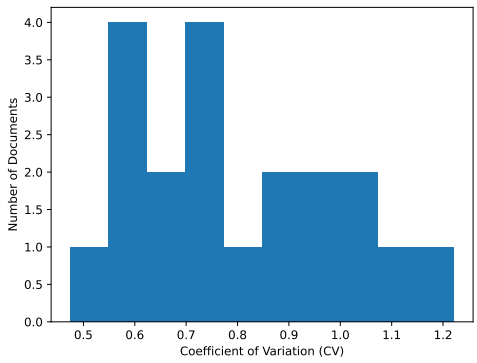

Figure 3: The coefficient of variation in scene lengths for each individual document in the training data.

this, we walk across the boundary candidates and, in a fixed-sized window, set the boundary class to zero for all but the largest value in the window. With a window size of five, for example, this means that no candidate with larger confidence values in its four neighbors (two to either side) will be predicted. Using this simple strategy, however, we adversely impact the quality of our predictions, going from an F1-Score of 33.7 to one of 27.8.

The improvements attained by application of the DP technique by Pethe et al. (2020) in combination with the variance of 0.74 in the task's trial document illustrate just how important non-local information is to improving performance in this task. Further work on neural sequence models may yield significant improvements.

Our final model uses the DP approach by Pethe et al. (2020) with $\alpha = 0.8$, a strong focus on local values. As explicitly stated in their paper, this method assumes knowledge of the actual number of boundaries, which is not the case for our data. We apply the heuristic of assuming the number of actual boundaries to be equal to the number of locally predicted boundaries. This way our the non-local approach effectively only moves the positions at which splits happen but does not change their total number. Unsurprisingly, given the variance in scene lengths, we found this to outperform the heuristic of dividing the text length by the average scene length. Further, we adapt the cost function to be more lenient with regard to scenes shorter than the average, as long as they are not too short.

Figure 4 shows how we adapt the equidistant constraint by Pethe et al. (2020) to punish very short distances. Where their cost function is linear in both directions, we adapt it to only punish very

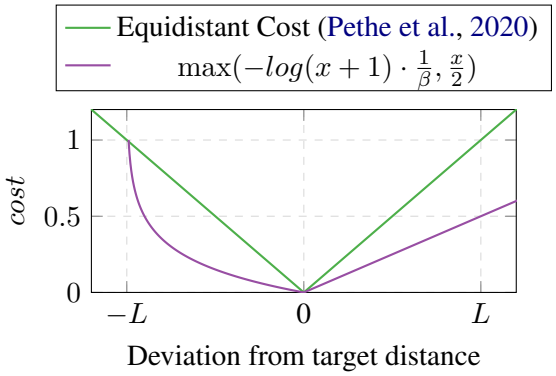

Figure 4: The cost associated with deviation from the target distance L, where a deviation of $-L$ is equivalent to a boundary distance of zero

short scenes harshly.

$$-log(x+1)\cdot\frac{1}{\beta} \quad (1)$$

For this, we apply the cost function in Equation 1 to negative distances relative to the target distance L, β is a hyperparameter controlling how close to a distance of zero very large costs set in; we use $\beta = 2$. For positive distances, we use $\frac{x}{2}$ effectively increasing the inherent α but also changing the relation of long distances to short ones.

Evaluating the same technique on our training data yielded a marginal improvement of around 0.01 F1, this is to be expected as some memorization of training samples should lead to improved local decisions. This result does give us confidence the approach will not adversely impact test set performance.

While, after optimizing alpha on the held-out data, the equidistant cost function performed on par with our cost function on the same data, when adapting to the training data (on which our α value was not optimized) the equidistant function only increased performance by 0.003 F1.

Further analysis is needed to provide a clear picture of cost function's impact on unseen data. It however already seems plausible that our adaptation of the cost function presents an improvement over the equidistant cost function.

6 Conclusion and Final Results

We present an approach to scene segmentation that relies on character information. While we do not produce irrefutable evidence of its advantages, we propose a cost function more suitable to the needs of scene segmentation, adapting the work by Pethe et al. (2020) to a new task.

On the official evaluation metric we only reach an F1-score of 0.02 for Track 1 and an F1-score of 0.11 for Track 2. These are below the boundary class performance discussed earlier as they include the correct classification of scene types. With out system focusing mostly on the placement of scene boundaries it could potentially be extended with features more suitable for scene classification.

The system performs relatively poorly in Track 1, reaching the last place with quite a margin to the next system, but much better in Track 2 where it is close behind the third-placed system, what exactly causes this difference in performance remains unclear. We stay far behind the performance of the top-scoring systems but coreference seems to be a salient feature that may be useful to include in future systems.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 2623–2631, Anchorage, Alaska, USA. Association for Computing Machinery.

James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. 2011. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems*, volume 24, pages 469–477, Granada, Spain. Curran Associates, Inc.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*, Online.

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, USA. Association for Computational Linguistics.

Dan Hendrycks and Kevin Gimpel. 2020. Gaussian error linear units (GELUs). *Computing Research Repository*, arxiv:1606.08415. Version 4.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python. https://github.com/explosion/spaCy/tree/v3.1.1.

Markus Krug, Lukas Weimer, Isabella Reger, Luisa Macharowsky, Stephan Feldhaus, Frank Puppe, and Fotis Jannidis. 2018. Description of a corpus of character references in German novels-DROC [Deutsches ROman Corpus]. *DARIAH-DE Working Papers*, 27:1–16.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. 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 687–692, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Charuta Pethe, Allen Kim, and Steve Skiena. 2020. Chapter Captor: Text Segmentation in Novels. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8373–8383, Online. Association for Computational Linguistics.

Fynn Schröder, Hans Ole Hatzel, and Chris Biemann. 2021. Neural end-to-end coreference resolution for German in different domains. In *Proceedings of the 17th Conference on Natural Language Processing*, Düsseldorf, Germany.

Albin Zehe, Leonard Konle, Lea Katharina Dümpelmann, Evelyn Gius, Andreas Hotho, Fotis Jannidis, Lucas Kaufmann, Markus Krug, Frank Puppe, Nils Reiter, Annekea Schreiber, and Nathalie Wiedmer. 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, Online. Association for Computational Linguistics.

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*.