Summarization-based Data Augmentation for Document Classification

Yueguan Wang
The University of Tokyo
etsurin@iis.u-tokyo.ac.jp

Naoki Yoshinaga
Institute of Industrial Science,
The University of Tokyo
ynaga@iis.u-tokyo.ac.jp

Abstract

Despite the prevalence of pretrained language models in natural language understanding tasks, understanding lengthy text such as document is still challenging due to the data sparseness problem. Inspired by that humans develop their ability of understanding lengthy text form reading shorter text, we propose a simple yet effective summarization-based data augmentation, SUMMaug, for document classification. We first obtain easy-to-learn examples for the target document classification task by summarizing the input of the original training examples, while optionally merging the original labels to conform to the summarized input. We then use the generated pseudo examples to perform curriculum learning. Experimental results on two datasets confirmed the advantage of our method compared to existing baseline methods in terms of robustness and accuracy. We release our code and data at https://github.com/etsurin/summaug.

1 Introduction

Although the pretrained language models (Devlin et al., 2019; Liu et al., 2019; He et al., 2020) have boosted the accuracy of various natural language understanding tasks, the accuracy is still limited for complex tasks with lengthy input (Lin et al., 2023) and fine-grained output (Liu et al., 2021), such as document classification. These tasks require models to find a mapping between diverse input and output, which models are more likely to suffer from the data sparseness problem.

To address the data sparseness problem, researchers have studied data augmentation for text classification tasks. A basic approach is to generate pseudo training examples from gold examples by perturbing the inputs; those perturbation include back-and-forth translation (Shleifer, 2019) and minor editing of input text (Wei and Zou, 2019; Karimi et al., 2021) or its hidden representations (Chen et al., 2020, 2022; Wu et al., 2022). These methods basically echo the information in the original training data, which will not help much the model learn to read lengthy inputs.

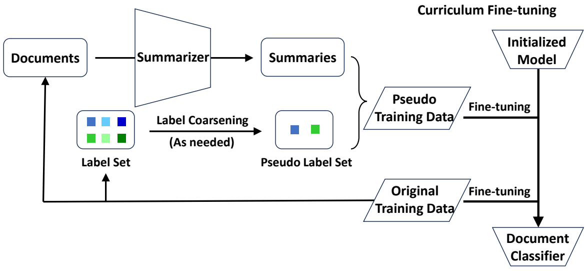

Figure 1: Curriculum fine-tuning for document classification using SUMMaug data augmentation: prior to the normal finetuning, it fine-tunes a model with easy-to-learn examples obtained by summarizing the original training examples.

In this study, to effectively develop the model's ability to comprehend the content in document classification, we propose a simple yet effective summarization-based data augmentation, SUMMaug, to generate pseudo, abstractive training examples for document classification. Specifically, we apply text summarization to the input of gold examples in document classification task to obtain abstractive, easy-to-read examples, and merge fine-grained target labels as needed so that the labels conforms to the summarized input. Motivated by that we humans gradually develop the ability of understanding lengthy text from reading shorter text, we use the generated examples in the context of curriculum learning (surveyed in (Soviany et al., 2022)), namely, curriculum fine-tuning.

We compare our method to a baseline data augmentation (Karimi et al., 2021) on two versions of IMDb dataset with a different number of target labels. Experimental results confirm that curriculum fine-tuning with SUMMaug outperforms baseline methods on both accuracy and robustness.

Proceedings of The 4th New Frontiers in Summarization Workshop, pages 49–55
December 6, 2023 ©2023 Association for Computational Linguistics

2 Related Work

In this section, we first review existing neural models for document classification, and next introduce existing data augmentation methods for text classification. We then mention other attempts to leverage summarization for text classification.

Document Classification In the literature, researchers explore a better neural architecture to comprehend the lengthy content in document classification; examples include a graph neural network (Zhang and Zhang, 2020; Zhang et al., 2022) and a convolutional attention network (Liu et al., 2021). Recently, Transformer (Vaswani et al., 2017)-based models have been revisited (Dai et al., 2022) and reported to outperform the task-specific networks. Since our work is model-agnostic and orthogonal to the model architecture, we adopt RoBERTa (Liu et al., 2019), a Transformer-based pre-trained model, as the target of evaluation.

Data Augmentation for Text Classification To address the data sparseness problem in text classification, researchers employ data augmentation, which generates pseudo training examples from the training examples. Shleifer (2019) leverages back-and-forth translation to paraphrase the inputs of training examples. Through translating the inputs into another language and then translating the resulting translation back to the source language, they obtain the input that are written in different ways but will have the same meanings conforming to the corresponding target labels. Xie et al. (2017) perturb the input by deleting and inserting words and replacing words with their synonyms. Karimi et al. (2021) propose a simple but more effective perturbation that randomly inserts punctuation marks. Rather than directly perturbing the input of training examples, some studies add noises in their continuous representations (Chen et al., 2020, 2022; Wu et al., 2022). However, these method predominantly echo existing training data, providing minimal assistance in understanding lengthy texts.

Use of Summarization in Text Classification Li and Zhou (2020) and Hartl and Kruschwitz (2022) utilize automatically generated summaries to retrieve fact for fake news detection. Whereas this approach uses summaries to retrieve knowledge for classification, our approach leverages summaries in training as easy-to-learn examples, which does not assume costly summarization in inference.

3 SUMMaug

Document classification requires a model to comprehend lengthy text with dozens of sentences, which is even difficult for humans, especially, children and second-language learners. Then, how do we humans develop an ability to comprehend lengthy text? In school, starting from reading short, concise text, we gradually read longer text.

In this study, we develop a summarization-based data augmentation method for document classification, SUMMaug, and use it to generate pseudo, abstractive training examples from gold examples to perform curriculum learning in document classification.

3.1 Summarization-based data augmentation

In SUMMaug, a summarization model M is used to generate pseduo, easy-to-learn examples for document classification. In this study, we apply an off-the-shelf summarization model, M, to each training pair $\{x, y\}$, where x denotes the document and y denotes the label, and then obtain a concise summary of x, namely, $\hat{x} = M(x)$.

An issue here is how to determine the label for the generated concise summary, $\hat{x}$. Since the summarization abstracts away detailed information for classification, the original target label y can be inappropriate especially when the target labels are fine-grained. We thus define a map function f to merge the fine-grained categories into a coarse-grained label group, and obtain the augmented training pair is $\{\hat{x}, f(y)\}$, as shown in Figure 1.

On summarization model To summarize diverse text handled in document classification, we assume an off-the-shelf summarization model that can handle documents with diverse topics. In this study, we choose an off-the-shelf BART (Lewis et al., 2020)-based summarization model fine-tuned on CNN-Dailymail (Hermann et al., 2015) dataset as an implementation of M,[1] since the writing style of news reports is suitable for most of the text in daily life. We should mention that the CNN-Dailymail dataset contains mostly extractive summaries, and the resulting summarization model will be less likely to suffer from hallucinations (Maynez et al., 2020) that have been reported for a summarization model trained on abstractive summarization datasets such as XSum (Narayan et al., 2018).

[1] https://huggingface.co/facebook/bart-large-cnn

I am Anthony Park, Glenn Park is my father. First off I want to say that the story behind this movie and the creation of the Amber Alert system is a good one. However the movie itself was poorly made and the acting was terrible. The major problem I had with the movie involved the second half with Nichole Timmons and father Glenn Park. The events surrounding that part of the story were not entirely correct. My father was suffering from psychological disorders at the time and picked up Nichole without any intent to harm her at all. He loved her like a daughter and was under the mindset that he was rescuing her from some sort of harm or neglect that he likely believed was coming from her mother who paid little attention to her over the 3 plus years that my father took care of her and summarily raised her so her mother could frolic about. The movie depicted my father in a manner that he was going to harm her in some way shape or form. The funny thing is that Nichole had spent many nights sometimes consecutively at my fathers place while Sharon would be working or doing whatever she was doing. The reason that my father was originally thought to be violent was because he had items that could be conceived to be weapons on his truck. My father was a landscaper. The items they deemed to be weapons were landscaping tools that he kept in his truck all the time for work. My recommendation is take this movie with a grain of salt, it is a good story and based on true events however the details of the movie (at least the Nichole Timmons - Glenn Park portion) are largely inaccurate and depict the failure of the director to discover the truth in telling the story. The funny thing is, that if the director would have interviewed any of Sharon's friends who knew the situation they would have stated exactly what I have posted here.
The movie itself was poorly made and the acting was terrible. The events surrounding that part of the story were not entirely correct. My recommendation is take this movie with a grain of salt, it is a good story and based on true events.

Table 1: An example of original text an generated summary on IMDb dataset. The first row is the original text while the second row is the generated summary. Red text are counterparts of summary in the original text.

Table 1 exemplifies a summary generated for IMDb datasets. While the original input (review) exhibits a mild negative sentiment, its compression into a summary intensifies this sentiment. This observation underscores the imperative to categorize labels of augmented data into coarser groups.

3.2 Learning a Classifier with augmented data

In the literature of data augmentation, the models are basically trained with the original and augmented training data, since both data are related to the target task. In our settings, however, the labels will be merged into fewer labels so that the labels conform to the generated summaries. We thus consider the following two strategies to utilize the pseudo abstractive training data.

Mixed fine-tuning We combine the original and pseudo training data to fine-tune a pre-trained model for classification. In this setting, we do not collapse labels, namely, $f(y) = y$.

Curriculum fine-tuning We first finetune a pre-trained model on the pseudo training data, and then finetune a pre-trained model on the original training data. This strategy is inspired by curriculum learning (Bengio et al., 2009). In this setting, we collapse labels as needed. When we collapse labels, we discard parameters for the collapsed labels in the fine-tuning with the original examples.

In the following experiments, we compare two strategies for datasets with different numbers of labels.

Dataset	train	val	test	C	L	L_M
IMDb-2	22500	2500	25000	2	279.5	51.3
IMDb-10	108670	13432	13567	10	394.2	50.2

Table 2: Details of the IMDb datasets: C denotes the number of classes. L and L_M denote the average length of the inputs and the generated summaries, respectively.

4 Experiments

We conduct experiments on two datasets to evaluate our method, thus demonstrating that: (1) our method shows better accuracy and robustness compared with baseline methods in both general setting and low-resource settings; and (2) curriculum fine-tuning plays an important role in achieving improvements.

4.1 Dataset

We use two versions of large-scale movie reviews dataset IMDb for evaluation. One contains 50,000 movie reviews with a positive or negative label (Maas et al., 2011), while the other involves 10 different labels from rating 1 to 10. For the IMDB-2 dataset, we split 10% of the training data for validation. For the IMDb-10 dataset, the same splitting as Adhikari et al. (2019) is used. The detailed information of the two datasets is shown in Table 2.

4.2 Methods

We use the following three models for evaluation. All models are based on RoBERTa (Liu et al., 2019) with a classification layer.

Model	The size of training data		
	200	1500	all
RoBERTa	92.19$_{1.21}$	94.21$_{0.62}$	94.63$_{0.56}$
+ AEDA (mixed)	90.91$_{1.44}$	94.43$_{0.49}$	94.75$_{0.66}$
+ AEDA (curriculum)	**93.59**$_{1.16}$	94.26$_{0.74}$	**95.56**$_{0.12}$
+ SUMMaug (mixed)	92.94$_{0.99}$	94.61$_{0.64}$	94.85$_{0.62}$
+ SUMMaug (curriculum)	93.36$_{0.97}$	**94.77**$_{0.28}$	95.45$_{0.17}$

Table 3: Classification accuracy$_{\text{stdev.}}$ (%) on IMDb-2: mixed and curriculum denotes mixed and curriculum fine-tuning. All the results are averages over five runs. The best results are marked as **bold**.

Model	The size of training data	
	1500	all
RoBERTa	39.99$_{8.46}$	56.58$_{0.34}$
+ AEDA (mixed)	36.58$_{10.64}$	51.23$_{14.39}$
+ AEDA (curriculum)	41.77$_{3.01}$	56.63$_{1.65}$
+ SUMMaug (mixed)	40.65$_{2.71}$	55.81$_{2.00}$
+ SUMMaug (curriculum)	**42.14**$_{1.48}$	**57.55**$_{0.29}$

Table 4: Classification accuracy$_{\text{stdev.}}$ (%) on IMDb-10. All the results are averages over five runs. The notations follow Table 3.

RoBERTa We finetune a pre-trained RoBERTa[2] on the original training data as a baseline.

RoBERTa + AEDA We use AEDA (Karimi et al., 2021), a strong data augmentation method for text classification as another baseline. We apply AEDA[3] to the original documents, and then fine-tune a RoBERTa model on the augmented data and original data.

RoBERTa + SUMMaug We use BART-based summarizer trained on CNN-Dailymail to generate concise summaries, and fine-tune a RoBERTa model on the augmented data and original data.

To evaluate the performance of our method in low resource settings, we randomly select 200 and 1500 samples from the two datasets and train a model on these sub datasets. However, on the IMDb-10 dataset, we observe that all models diverge and perform randomly when training data is reduced to 200, likely due to the challenges of fine-grained classification with rather limited training data; we thereby do not report the results.

In order to reveal the effectiveness of curriculum fine-tuning, we apply curriculum fine-tuning not only to SUMMaug but also to AEDA. On the IMDb-10 dataset, we map the labels of the augmented data into coarse-grained ones, as mentioned in § 3.1. Specifically, labels between 0-4 are mapped into 0 (negative) while labels between 5-9 are mapped into 1 (positive).

4.3 Implementation Details

We set the model's hyperparameters as follows. For experiments on the IMDb-2 dataset, batch size is set to 64 and learning rate is set to 1e-5. For experiments on the IMDb-10 dataset, following Adhikari et al. (2019), batch size is set to 16, with learning rate set to 2e-5. Detailed information of training epochs can be found at Appendix A. All the experiments were conducted on four NVIDIA Quadro P6000 GPUs with 24GB memory.

The final model for evaluation is selected on the basis of the performance on validation set. To eliminate the effect of random factors, we report the average accuracy over five runs.

5 Results

Tables 3 and 4 list the results of baseline methods and our proposed method. Our method outperforms baseline methods in all experimental settings. We additionally confirm on both datasets that our data augmentation is effective even when the training data size is small.

How robustly does SUMMaug work? SUMMaug achieves higher classification accuracy across datasets while improving or maintaining robustness (low standard deviations), whereas the original AEDA, namely AEDA (mixed), reduces the accuracy on IMDb-2 when 200 training examples are used, and it leads to unstable results on IMDb-10 dataset.

Is curriculum fine-tuning effective? We use mixed fine-tuning with SUMMaug and curriculum fine-tuning with AEDA. We observe that under mixed fine-tuning method, the data augmented by SUMMaug exhibited less improvements and even turns to be harmful on the IMDb-10 dataset. Conversely, it turns out that curriculum learning helps the AEDA method achieve further improvements in some cases while addressing the low robustness issue. However, curriculum learning with AEDA does not consistently enhance results because the AEDA augmented data retains the same information as the original data, which offers limited benefits in improving text comprehension.

[2] https://huggingface.co/roberta-large

[3] https://github.com/akkarimi/aeda_nlp

N	f	Accuracy$_{\text{stdev.}}$
2	[0,0,0,0,0,1,1,1,1,1]	$57.55_{0.29}$
3	[0,0,0,1,1,1,1,2,2,2]	$57.47_{0.20}$
4	[0,0,0,1,1,2,2,3,3,3]	$57.66_{0.31}$
5	[0,0,1,1,2,2,3,3,4,4]	$57.32_{0.60}$
10	[0,1,2,3,4,5,6,7,8,9]	$57.20_{0.56}$

Table 5: Classification accuracy$_{\text{stdev.}}$ (%) on IMDb-10 with different label coarsening function under SUMMaug (curriculum) method. N denotes the number of merged label groups while f shows how the original label 0-9 is mapped into coarse-grained label. All the results are averages over five runs.

How label coarsening affects accuracy? Table 5 shows the results of SUMMaug (curriculum) under different map function f. The accuracy is comparable when $N \leq 4$, while there's a noticeable decline in accuracy, accompanied by decreased stability when label coarsening is insufficient or not adopted. This is probably because the summaries can filter out detailed content, which is essential for fine-grained classification. On the other hand, unlike mixed fine-tuning, in which potentially noisy augmented data is used throughout the training process, in the curriculum fine-tuning, the effect of noise diminishes after model turns to train on the original data. Consequently, it can still achieve improvement even without label coarsening.

6 Conclusion and Future Work

This study explores a novel application of a summarization model and proposes a simple yet effective data-augmentation method, SUMMaug, for document classification. It performs curriculum learning-style fine-tuning to first train a model on concise summaries prior to the fine-tuning on the original training data. This mirrors the human process of mastering lengthy text comprehension, through gradual exposure to longer text. Experimental results on two document classification datasets confirm that SUMMaug enhances both accuracy and training stability compared to the baseline data augmentation method. Meanwhile, our method shows effective in low-resource settings.

The future work will focus on searching for the optimal mapping function f and exploring the effect of different summarization models. We will also apply SUMMaug to other document classification tasks of various domains.

Limitations

One of the drawbacks of this study is that we do not consider the label coarsening function f as a hyperparameter and just choose the simplest one for experiments. The effect of label coarsening function on accuracy is still insufficiently explored. For the datasets, despite the different numbers of labels, the documents used are originally from the same kind of domain, which is not convincing enough to show that SUMMaug is robust across diverse classification tasks in different domains.

Acknowledgements

This work was partially supported by the special fund of Institute of Industrial Science, The University of Tokyo, and by JSPS KAKENHI Grant Number JP21H03494.

References

Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2019. DocBERT: BERT for document classification. *arXiv preprint arXiv:1904.08398*.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 41–48, New York, NY, USA. Association for Computing Machinery.

Hui Chen, Wei Han, Diyi Yang, and Soujanya Poria. 2022. DoubleMix: Simple interpolation-based data augmentation for text classification. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4622–4632, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Xiang Dai, Ilias Chalkidis, Sune Darkner, and Desmond Elliott. 2022. Revisiting transformer-based models for long document classification. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7212–7230, Abu Dhabi, United Arab Emirates. 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, Minneapolis, Minnesota. Association for Computational Linguistics.

Philipp Hartl and Udo Kruschwitz. 2022. Applying automatic text summarization for fake news detection. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2702–2713, Marseille, France. European Language Resources Association.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced BERT with disentangled attention.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28.

Akbar Karimi, Leonardo Rossi, and Andrea Prati. 2021. AEDA: An easier data augmentation technique for text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2748–2754, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Qifei Li and Wangchunshu Zhou. 2020. Connecting the dots between fact verification and fake news detection. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1820–1825, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Yu-Chen Lin, Si-An Chen, Jie-Jyun Liu, and Chih-Jen Lin. 2023. Linear classifier: An often-forgotten baseline for text classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1876–1888, Toronto, Canada. Association for Computational Linguistics.

Yang Liu, Hua Cheng, Russell Klopfer, Matthew R. Gormley, and Thomas Schaaf. 2021. Effective convolutional attention network for multi-label clinical document classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5941–5953, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Sam Shleifer. 2019. Low resource text classification with ULMFit and backtranslation. *arXiv preprint arXiv:1903.09244*.

Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. 2022. Curriculum learning: A survey. *International Journal of Computer Vision*, 130(6):1526–1565.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Xing Wu, Chaochen Gao, Meng Lin, Liangjun Zang, and Songlin Hu. 2022. Text smoothing: Enhance various data augmentation methods on text classification tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 871–875, Dublin, Ireland. Association for Computational Linguistics.

Ziang Xie, Sida I. Wang, Jiwei Li, Daniel Lévy, Aiming Nie, Dan Jurafsky, and Andrew Y. Ng. 2017. Data noising as smoothing in neural network language models. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April*

Dataset (fine-tuning method)	The size of training data		
	200	1500	all
IMDb-2 (w/o data augmentation)	70	18	2
IMDb-2 (mixed)	70	18	2
IMDb-2 (curriculum)	70/70	18/18	2/2
IMDb-10 (w/o data augmentation)	-	20	4
IMDb-10 (mixed)	-	20	4
IMDb-10 (curriculum)	-	5/20	2/6

Table 6: Detailed training epochs in our experiments. For curriculum fine-tuning method, x/y denotes that model is trained x epochs on augmented data and then y epochs on original data.

24-26, 2017, Conference Track Proceedings. OpenReview.net.

Chong Zhang, He Zhu, Xingyu Peng, Junran Wu, and Ke Xu. 2022. Hierarchical information matters: Text classification via tree based graph neural network. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 950–959, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Haopeng Zhang and Jiawei Zhang. 2020. Text graph transformer for document classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8322–8327, Online. Association for Computational Linguistics.

A Detailed training epochs

Table 6 shows detailed training epochs in our experiments. We select training epochs based on the accuracy on the validation set.