Reducing Processing Time and Enhancing Classification Performance: Shortening Strategies for German Public Contributions

Anonymous ACL submission

Abstract

Public participation, the process of voluntary engagement of citizens in urban decision-making, requires a significant amount of time and human resources. Therefore, automatization of the evaluation is essential. Classification of the citizens' proposals is one of the prevalent analytical tasks. Through the years many studies have worked on automatization techniques of this procedure and Natural Language Processing (NLP) methods are among the most effective ones. Nevertheless, most developed techniques despite promising results are optimized for the English language. Moreover, the NLP pre-trained models such as BERT have limitations in the length of the texts they can process. Hence, this paper focuses on the abstractive summarization of the public proposal in German and considers two different shortening techniques (truncation and summarization). The main aim is to explore how pre-trained models such as the BERT perform in the classification of summarized German language contributions. For this purpose, the German BERT model, which is fine-tuned on the MLSUM DE dataset, and the multilingual BART model are considered for text summarization. The results revealed that applying shortening techniques on long contributions reduces the model development time by an average of 48% on CPU and 36% on GPU while improving performance. Moreover, the multilingual BART model works slightly better than the BERT model fine-tuned on the MLSUM DE dataset.

1 Introduction

Public participation, an attractive topic for academics, regulators, and governments, is one of the key instruments of democracy. Public contribution helps citizens' concerns and thoughts to be heard and avoids elitism (Rowe and Frewer, 2004). Even though consideration of the public voice provides more information to the authorities, it requires a longer processing time (Lee and Kim, 2014). Indeed, processing and analyzing the collected data is among the key challenges that policy-makers in the public voice consideration face (Romberg and Escher, 2023, Simonofski, Fink, and Burnay, 2021, Arana-Catania et al., 2021). To protect the ultimate goals of public participation and the core standards of a democratic society, all contributions should be treated equally. This emphasizes the requests for specific analytical methods to ease inferring meaningful information from an overwhelming collection of public statements (Lee and Kim, 2014, Romberg, Mark, and Escher, 2022, Romberg and Escher, 2022).

Previous experiences have demonstrated that when the amount of data is manageable, manual processing despite its inefficiency is preferred (Simonofski, Fink, and Burnay, 2021). However, the advances in technology especially in Artificial Intelligence (AI) and particularly Natural Language Processing (NLP) have made a huge difference (Romberg and Escher, 2023). NLP analyses linguistic data and enables computers to understand, interpret, and generate human language (Egger and Gokce, 2022). Despite notable advances in NLP techniques, such as the rise of Pre-trained Language Models (PLMs), Romberg and Escher (2023) in a review of methods for computational text analysis remark that public contribution texts exhibit distinct characteristics compared to other domains such as news or social media contents (Romberg and Escher, 2023). Another significant gap in the study of public contributions is language. The majority of the existing research is limited to English, with only a

few addressing non-English public proposal texts (Romberg and Escher, 2023, Romberg, 2023, Balta et al., 2019, Romberg and Conrad, 2021, Romberg, 2022). Hence further studies are required to generate more robust and reliable models.

Automatic Text Summarization (ATS) is another field of NLP, which despite its high potential has been left under-explored, especially in the study of non-English texts (Aruneshwari et al., 2024, Alcantara et al., 2023). ATS is a method for generating a coherent shorter version of a text document while preserving the key information from the original text (Aruneshwari et al., 2024). ATS is classified into two categories. (1) The abstractive text summarization, in which the system tries to understand the text and provides a shorter novel narrative of the original text while preserving the key points (Alcantara et al., 2023). (2) The extractive summarization, which is more flexible where the system first identifies the key sentences (texts), then extracts, orders, and returns them in a condensed form (Alcantara et al., 2023, Verma and Verma, 2020, Talib, 2021).

One of the main applications of text summarization is in the classification of the texts. However, this area is in its early stages and the majority of the studies are still focused on the English language. (Scialom et al., 2020) mentions that the dominance of the English language, lack of multilingual data, and the use of pre-trained models on the pivot language (English) resulted in a significant gap in the performance of the classification studies. To the best of our knowledge, this is the first study in summarization of public participation in the German language. Our contribution is mainly on understanding how summarization affects the models' performance in detecting argumentation structures.

The rest of this paper is organized as follows. Section 2 highlights the importance of analyzing public participation and mentions the previously explored techniques. This section also briefly reviews the Automatic Summarization techniques and models. Section 3 introduces the dataset. Section 4 comprehensively explains the developed methods in this study. In Section 5, the obtained results are presented and are discussed in Section 6. Finally, Section 7 summarizes the findings and mentions the topics for further studies.

2 Literature Review

2.1 Evaluation of Public Participation

Public participation, alternatively called citizen participation, is the volunteer action of citizens to influence the decision-making processes (Schroeter et al., 2016). However, processing all the collected statements to infer meaningful information manually, while respecting the principles of democracy is a challenging task (Karic et al., 2024, Arana-Catania et al., 2021). Hence, to achieve this goal (Kwon et al., 2006), (Arguello et al., 2008), (Habernal & Gurevych, 2016), and (Cardie et al., 2008) applied SVM models, (Konat et al., 2016) considered graph-based analytics, and with the progress of AI-based technologies in recent years such as Natural Language Processing (NLP) techniques, (Fierro et al., 2017) considered fastText and Deep Averaging Networks (DAN) to automatically analyze or classify a large number of public comments and arguments on proposed regulations. (Liebeck et al., 2016) is one of the few researchers in this domain who developed a new German corpus for argument mining and compared the performance of SVM, Random Forest (RF), and k-Nearest-Neighbor (kNN) classification models. (Romberg and Escher, 2022) studied in the domain of argument mining in German introduced a new dataset of German language arguments and (Romberg, Mark, and Escher, 2022) compared the performance of SVM, Maximum Entropy (MaxEnt), Naïve Bayes (NB), and BERT classifiers in single and multi-label classification of contributions.

2.2 Automated Text Summarization

Text Summarization has been always an interesting topic for researchers. In 2010 with the introduction of neural networks and later the development of Sequence-to-Sequence (Seq2Seq), Bidirectional Encoder Representations from Transformers (BERT), and Generative Pre-trained Transformer (GPT) models, more attention was attracted to automatic text summarization (Khan et al., 2023). Pre-trained models have significantly improved NLP capabilities (Alcantara et al., 2023). BERT is one of the bidirectional models pre-trained over texts from Wikipedia and has shown relatively good performance in abstractive text summarization. Some other known encoder and decoder pre-trained models are BART developed by Facebook, GPT developed by OpenAI, and

RoBERTa and DistilBERT both as a refined version of BERT (Alcantara et al., 2023, Syed, Gaol, & Matsuo, 2021). Chen et al. (2021) by highlighting the huge size of famous pre-trained models such as GPT and their extensive computation cost, tried to save the training cost of large models by transferring the learned knowledge of the t smaller rained model to the large model (Chen et al., 2021). They called the model BERT2BERT.

Pre-trained models despite significant performance have some limitations. The majority of them such as BERT are trained over English language documents which leads to lower performance on non-English documents. In order to overcome this limitation, multilingual versions of some of the models are introduced. mBERT, the multilingual version of BERT, is trained for the top 100 languages with the longest documents on Wikipedia (GitHub, 2022). This model supports the German language. German BERT2BERT fine-tuned on MLSUM DE for summarization is another pre-trained BERT model for the German language trained on German Wikipedia, OpenLegalData, and news articles. The BERT2BERT model is specialized for text summarization and is accessible through the Hugging Face website API for pre-trained models. MLSUM is the first large-scale MultiLingual SUMmarization dataset introduced in 2020 (Scialom et al., 2020). The dataset contains over 1.5 million article/summary pairs in five languages, including German, and can be used to evaluate the summarization models (Scialom et al., 2020).

2.3 Research Gap

Many studies have sought to reduce the burden of analyzing public contributions through text-processing techniques. However, most of these studies have focused on the English language, and there is a lack of exploration into the fine-tuning of transformer models (such as BERT) and the performance of other transformer architectures (Romberg, Mark, and Escher, 2022). This study aims to address these gaps and analyze the potential limitations of applying developed models to long texts. As a potential solution, automatic text summarization techniques, whose performances on argumentative data in the German language are largely unexplored (Alcantara et al., 2023), are tested and evaluated. The primary goal of this study is to assess how summarization impacts runtime and overall performance.

	Single Label			Multi-Label		
	Bonn	**Ehrenfeld**	**Moers**	**Bonn**	**Ehrenfeld**	**Moers**
Bicycle parking	108	22	9	112	26	9
Cycle path quality	449	58	111	519	71	118
Cycling traffic management	1020	195	222	1056	204	229
Lighting	37	1	10	47	2	15
Misc	53	5	10	84	84	27
Obstacles	319	35	31	364	45	33
Signage	150	16	19	182	20	27
Traffic lights	178	34	47	197	39	51
Total	**2314**	**366**	**459**	**2314**	**366**	**459**

Table 1: Distribution of the arguments across labels

3 Dataset

The dataset considered for this study is a collection of public contributions from three different cities in Germany and has been used for automated topic classification (Romberg and Escher, 2022). The public participation processes, known as "Raddialoge", were held by the municipalities of Bonn, Ehrenfeld, and Moers from September to October 2017. Detailed information on the collection of contributions and labeling can be found in (Romberg and Escher, 2022).

The total number of collected contributions is 3139 of which 2314 were received from Bonn, 366 from Ehrenfeld, and 459 from Moers. Eight different categories for improving the citizens' experience were proposed and each contribution was assigned to one or more of these categories. An overview of the dataset is given in Table 1. In total, there are 231 contributions from Bonn, 46 contributions from Moers, and 36 contributions from Ehrenfeld which belonged to two categories. Of the defined categories, "Cycling Traffic Management" and "Cycle Path Quality" have received the most contributions. As claimed by (Romberg and Escher, 2022), the variation in the distribution of contributions is affected by factors such as city size, local infrastructures, and participators' involvement.

4 Experimental Setup

In this section, we provide the applied techniques and classification algorithms. As is well known the

uneven distribution of the contributions over categories can potentially degrade the classification performance. Hence, weighting techniques are tested as a possible solution to improve the results. For the classification of the multi-label contributions, to improve the performance, in addition to the class weights, the Area Under Curve (AUC) score is used to evaluate the model's performance across different thresholds. The thresholds are applied to the output of the model to determine the final decision on the predicted class of a contribution. Experiments are done using the Google Colab GPU (T4), Python programming language, and the software provided by Hugging Face, Inc. (Wolf et al., 2020). The Transformers library of Hugging Face eases the efforts for a variety of NLP tasks.

4.1 BERT model

The first applied classification model is based on the BERT model initialized with the case-sensitive *gbert-base*. BERT is a pre-trained model specialized for the tokenization and encoding of the German language. However, it is fundamentally an encoder-only model that cannot be directly used for classification tasks. Thus, BertForSequenceClassification which provides a classifier on top of the BERT model is used. BertForSequenceClassification is a class from the HuggingFace Transformers library and lets us fine-tune the pre-trained BERT model on a specific sequence classification task. The results obtained from this stage are considered a baseline for this experiment.

4.2 German BERT2BERT Fine-Tuned on MLSUM DE model

This model is based on the German BERT, initialized with the bert-base-german-cased model and fine-tuned on the MLSUM DE dataset. It is specifically developed for German text summarization. After summarizing the citizens' proposals, the BertForSequenceClassification is used for classification.

4.3 BART Model

BART is a transformer-based model that generalizes BERT (encoder) and GPT (decoder) models. It is pre-trained and combines Bidirectional and Auto-Regressive Transformers, and is particularly effective in text generation or text comprehension tasks (Lewis et al., 2020). In this study, the performance of the model specifically initialized for the German language *(gbert-base)* is compared to a multilingual model, "*facebook/mbart-large-50*". This model is Sequence-to-Sequence and supports the German language. The summarized contributions are classified using the BertForSequenceClassification.

4.4 Implementation

Transformers, despite the notable improvements brought to the NLP world, experience a significant computational overhead due to their attention mechanism. Indeed, from the autoregressive nature of the attention mechanism, the complexity depends on the quadratic of the sequence size ($O(T^2)$) (Cukier, 2024). This dependency forces a limit on the sequence length that can be passed to the models. On the other hand, the extended length of training sequences can result in training inefficiency. From the nature of transformer models, all the sequences of input texts need to be equal in length. This necessitates padding of the shorter documents while padding adds no meaningful information, and their processing wastes computational resources.

The majority of the transformer-based models can handle up to 512 or 1024 tokens (Hugging Face, 2024). The maximum length that the BERT model supports is 512. Making decisions based on the maximum length can affect the classification model performance. Different studies have proposed different strategies to overcome this limitation. One of the naïve solutions is truncation of the longer sequences. However, studies have shown that different parts of texts carry different amounts of information. (Chi et al., 2019) in a study on English texts found that keeping the head and the tail of a text and dropping the middle part gives the best performance. In another study, Mutasodirin and Prasojo (2021) compared the performance of truncation and summarization techniques in the classification of texts from the Indonesian News Article dataset (IndoSum). They stated that using the first block of the tokens results in the best performance in comparison to the other truncation and summarization techniques (Mutasodirin and Prasojo, 2021).

The motivation for this study specifically lies in the tokenization stage of the contributions. Table 2 summarizes the statistics of the contribution

lengths after tokenization. The numbers show that the distribution of the contribution lengths among the cities is very close. Moreover, almost 90% of the contributions in all of the cities have less than 128 tokens. This number is half of the maximum length applied by (Romberg and Escher, 2022) as the hyper-parameters of the BERT model. Choosing 256 as the maximum length while the longest contribution has 247 tokens positively avoids truncation and losing information. However, more than 90% of the texts get padded to at least twice their length. This can potentially decrease the classification performance and increase the run time unnecessarily. In the following experiments, the performance of truncated and summarized texts is compared.

In the first experiment, in order to have a baseline for the rest of the experiments, the full-text contributions are classified using the BertForSequenceClassification classifier, and the BERT model is initialized with the case-sensitive '*gbert-base'* model. In the tokenization stage of the BERT model, the maximum length is set to 256 and the rest of the hyper-parameters of the model are kept unchanged.

In order to make the results more reliable, a 5-fold cross-validation is conducted. Thus, after the preparation step, a new set of stratified 5 folds is generated to ensure the same label distribution in both the training and testing data. However, for the city of Ehrenfeld, since the number of contributions for the category of "lighting" is only 1, stratified sampling fails. To overcome this limitation, we applied synonym replacement, an NLP data augmentation technique, and generated 4 new contributions by randomly replacing some of the words from the original contribution (Ma, 2019). For synonym replacement, a BERT pre-trained model initialized with '*deepset/gbert-base*' is chosen. Reported results in the following sections are averaged across all folds.

The issue of unbalanced label distribution in the dataset was addressed in Section 3. For instance, in Bonn, the number of proposals under the category of "cycling traffic management" is 27 times more than the lightning proposals. This condition exists also in the other two cities and among the other categories. Although such a distribution of the labels in the analysis of argument data is expected, it can negatively affect the performance of the model in the prediction of the minority classes or lower its generalization. One of the solutions is using class weights. In the second experiment, $W_i = \frac{1}{N_i} / \sum_{i=1}^{C} \frac{1}{N_i}$ weighting strategy is applied. In this formula, N_i is the number of samples under the i-th category, w_i is the assigned weight to the i-th category, and C is the total number of classes. The weights are applied to the loss function to penalize the model for the misclassification of the minority class.

	Bonn	Ehrenfeld	Moers
Mean	67.82	62.5	65.7
Minimum	3	6	4
Q1	34	28	33
Q2	56	51	54
Q3	89	82	89
P90	132	119	123
Maximum	247	231	238

Table 2: Basic statistics of the tokens' distribution

The third and fourth experiments are a repetition of the baseline while in the tokenization stage of the BERT model, the maximum length is set to 128 and the weighting strategy is applied, respectively. The fifth experiment tests the abstractive summarized texts generated using the multilingual BART (mBART) initialized with "*facebook/mbart-large-50*" model. Since any form of document shortening results in the loss of information, only the citizens' contributions with more than 128 tokens (based on white-space tokenization) are summarized. From the hyper-parameters, the minimum and maximum lengths are set to 50 and 120, respectively. The decision on the minimum and maximum lengths is based on the statistics from Table 2 to balance the need for reducing the size while keeping the summary concise. From Bonn, approximately 11% (250), from Ehrenfeld 9% (34) of contributions, and from Moers 9% (43) of the contributions are summarized. The average length of the summarized contributions is 64 for Bonn, 65 for Ehrenfeld, and 63 for Moers.

The sixth experiment is the repletion of the fifth experiment with class weights applied. The seventh experiment compares the performance of the German BERT2BERT summarization model in text classification against the multilingual mBART summarization model. The BERT2BERT model is initialized with *mrm8488/bert2bert_shared-german-finetuned-summarization*, using the earlier mentioned hyper-parameters, and is only applied on texts with more than 128 tokens. The average length of the summarized texts for Bonn, Ehrenfeld, and Moers are 39, 53, and 55,

respectively. It should be noted that although the minimum length hyperparameter of the models was set to 50, the best summaries may still be shorter depending on the influence of the other hyperparameters. More details on the distribution of the contributions' length after summarization are provided in Appendix B. Finally, the eighth experiment measures the performance of the sixth model when class weighting is applied.

At the end of the next section, all experiments are repeated to classify the multi-labeled contributions under two different scenarios. In the first scenario, the applied threshold on the models' output is set to 0. In the second, however, the AUC score is used to evaluate the model's performance across different thresholds. The threshold with the highest AUC score on the train data across all the folds of the cross-validation is considered as the best candidate.

5 Results and Evaluation

In this section obtained results for the earlier described cases are presented. Table 3 shows the category-wise performance of the trained models using the F-score. Reported results are the average of the outcome of testing the model on the test sets from the 5-fold cross-validation. The best model (with the highest micro F-score) for each city is highlighted.

For the city of Bonn, the first model (the baseline) successfully classified 78% of the arguments. The best and worst performances were obtained for categorizing "Bicycle Parking" and "Misc" labels. Experiment 2 shows the significant effect of class weights in the classification of the imbalanced data. While the overall performance (Micro F-score) has decreased by less than 1%, the classification performance of the least frequent labels "Lightning" and "Misc" have improved by 4% and 20%, respectively. Experiment 3 shows closely the same performance as the first two experiments. Nevertheless, since the contributions were truncated and only the first 128 tokens were considered, the main achievement is the 50% decrease in train time. Experiment 4 improved the performance of the third experiment, attained the best result, and the micro and macro f-scores surpassed the other experiments. Experiments 5 to 8 show the performance of the summarized texts. As it can be seen from the results, summarized texts using the multilingual model (mBART) were more successful. The main reason is in the retained

	Experiments	Bicycle Parking	Cycle Path Quality	Cycling Traffic Management	Lighting	Misc	Obstacles	Signage	Traffic Lights	Micro F-score	Macro F-score	Train run-time(second) GPU	Train run-time (second) CPU
Bonn	1_B	0.94	0.77	0.81	0.77	0.21	0.75	0.61	0.80	0.78	0.71	438	18480
	2_B	0.93	0.76	0.80	0.81	0.42	0.73	0.59	0.80	0.77	0.73	435	18490
	3_T	0.91	0.78	0.80	0.82	0.27	0.74	0.59	0.77	0.77	0.71	252	9185
	4_T	**0.92**	**0.77**	**0.81**	**0.79**	**0.52**	**0.75**	**0.63**	**0.81**	**0.78**	**0.75**	**236**	**9385**
	5_S	0.94	0.78	0.81	0.76	0.34	0.75	0.59	0.81	0.78	0.72	207	9140
	6_S	0.92	0.76	0.80	0.81	0.47	0.75	0.61	0.79	0.77	0.74	213	9010
	7_S	0.91	0.77	0.80	0.85	0.15	0.73	0.54	0.78	0.76	0.69	213	9010
	8_S	0.90	0.77	0.79	0.80	0.40	0.72	0.60	0.78	0.76	0.72	221	9025
Ehrenfeld	1_B	0.87	0.45	0.78	0.80	0.00	0.56	0.00	0.10	0.68	0.45	71	2882
	2_B	0.68	0.31	0.63	0.80	0.00	0.43	0.22	0.39	0.54	0.44	77	2941
	3_T	0.83	0.51	0.77	0.00	0.00	0.52	0.00	0.00	0.67	0.33	52	1492
	4_T	0.73	0.36	0.62	0.93	0.00	0.43	0.29	0.34	0.52	0.46	49	1497
	5_S	**0.80**	**0.56**	**0.80**	**0.40**	**0.00**	**0.56**	**0.00**	**0.16**	**0.70**	**0.41**	**51**	**1508**
	6_S	0.78	0.35	0.61	0.93	0.00	0.47	0.17	0.35	0.52	0.46	63	1502
	7_S	0.87	0.53	0.78	0.40	0.00	0.59	0.00	0.05	0.69	0.40	47	1542
	8_S	0.75	0.37	0.65	0.93	0.00	0.47	0.30	0.37	0.56	0.47	59	1545
Moers	1_B	0.63	0.77	0.82	0.00	0.00	0.29	0.21	0.84	0.75	0.45	416	4004
	2_B	0.63	0.66	0.70	0.00	0.00	0.36	0.33	0.78	0.65	0.43	433	4248
	3_T	**0.53**	**0.79**	**0.85**	**0.00**	**0.00**	**0.31**	**0.26**	**0.84**	**0.77**	**0.45**	**264**	**2000**
	4_T	0.92	0.75	0.82	0.13	0.00	0.47	0.43	0.80	0.76	0.54	233	1949
	5_S	0.47	0.78	0.83	0.00	0.00	0.30	0.00	0.83	0.75	0.40	249	2010
	6_S	0.83	0.75	0.78	0.26	0.00	0.34	0.30	0.76	0.72	0.50	244	1965
	7_S	0.47	0.76	0.83	0.00	0.00	0.20	0.24	0.80	0.74	0.41	248	2070
	8_S	0.81	0.75	0.80	0.00	0.00	0.31	0.44	0.73	0.73	0.48	255	2030

Table 3: Performance of the developed classification models on single-labeled contributions. Subscripts of **B**, **T**, and **S** represent Baseline, Truncation, and Summarization, respectively.

information. As noted earlier the average length of the summarized contributions using the BERT2BERT model is 39 for the city of Bonn. Undoubtedly this is a significant reduction in the contribution texts' length and hence, loss of information. Moreover, experiments 6 and 8 despite better performance in the classification of rare labels, because of decrements in the classification of frequent labels, show almost similar performance to experiments 5 and 7.

For Ehrenfeld, the results are different. In the first experiment, the model failed to classify contributions under the "Signage" and "Misc" labels. The model in the third experiment which was trained on the truncated contributions failed to classify the contributions of half of the categories. Nevertheless, it showed a close performance to that of the first model. The main reason for the

insignificant effect of classification failures on the micro f-score is due to the low number of samples under the failed categories. Furthermore, reducing the length of contributions by half has significantly decreased the train time. The best performance is achieved for the fifth experiment where the model is trained on the summarized texts from the multilingual model. The rest of the trained models show that applying weighting does not improve the overall performance of the models. In fact, weighting led to model confusion by affecting the decision boundaries and only enhanced the model's performance in classifying "Signage" and "Traffic Lights" labels. Finally, it should be noted that the applied data augmentation technique in the "lighting" class was successful. However, the result should be considered carefully as all the samples have the same characteristics.

Lastly, for Moers, the results show the same pattern as for the other two cities. Truncation and summarization significantly decreased train time while the former performed slightly better in the classification task. Weighting improved macro f-scores by enhancing the model's performance in the classification of rare categories, while micro f-scores decreased.

Table 4 provides detailed information on the performance of the train models on the multi-labeled contributions. The general behavior of the models is similar to the trained ones on the single-labeled contributions, and the changes in the training time of the models are in line with the former findings. Shortening reduced the run time by almost 47% for experiments conducted using a CPU compared to 37% for on GPU experiments.

In addition to the discussed metrics, as in the multi-label classification tasks, simultaneous prediction of multiple labels is probable; the Hamming Loss metric is typically used instead of accuracy. The hamming loss shows the percentage of the mislabeled arguments. For Bonn, experiments have labeled approximately 6% of the public contributions wrongly. For Ehrenfeld and Moers, however, the difference in the percentage of mislabeled contributions between the models with and without adjusted weights is significant. Weighting has lowered the models' performance under frequent classes in favor of less frequent ones. The results in Table 4 indicate that, aside from the failure of no-weighting models, the considered weighting technique also failed to enhance the models' performance in most experiments.

		Multi-labeled without AUC applied					Multi-Labeled with AUC applied				
	Experiments	Micro F-score	Macro F-score	Hamming (%)	Train time (sec.) GPU	Train time (sec.) CPU	Micro F-score	Macro F-score	Hamming (%)	Train time (sec.) GPU	Train time (sec.) CPU
Bonn	1_B	0.77	0.69	6	384	17482	0.76	0.73	7	430	22606
	2_B	0.75	0.71	6	383	18378	0.75	0.76	7	427	22126
	3_T	0.77	0.69	6	219	9955	0.76	0.73	7	336	11529
	4_T	0.75	0.71	6	221	8940	0.75	0.72	7	301	12300
	5_S	0.77	0.69	6	220	9935	0.75	0.72	8	256	13814
	6_S	0.75	0.71	7	225	9702	0.74	0.69	8	269	12743
	7_S	0.76	0.67	6	234	9120	0.74	0.71	8	329	13791
	8_S	0.74	0.69	7	233	10128	0.73	0.71	8	252	12164
Ehrenfeld	1_B	0.72	0.50	7	94	2782	0.63	0.44	11	71	3877
	2_B	0.57	0.24	12	93	2930	0.40	0.30	28	71	4100
	3_T	0.73	0.48	7	63	1635	0.64	0.44	10	45	2219
	4_T	0.55	0.18	12	65	1669	0.36	0.30	29	48	2290
	5_S	0.73	0.47	7	47	1750	0.63	0.41	11	44	1801
	6_S	0.55	0.19	12	46	1534	0.40	0.31	26	46	1765
	7_S	0.70	0.44	8	48	1517	0.63	0.43	10	50	2056
	8_S	0.55	0.18	12	46	1619	0.41	0.33	28	52	1985
Moers	1_B	0.71	0.31	7	85	4406	0.67	0.48	9	85	5449
	2_B	0.39	0.18	12	85	3730	0.50	0.39	16	87	4820
	3_T	0.73	0.31	7	57	1987	0.67	0.49	1	54	2800
	4_T	0.40	0.18	12	77	1837	0.47	0.42	20	55	2605
	5_S	0.72	0.31	7	54	2099	0.66	0.52	10	60	2770
	6_S	0.40	0.17	12	55	1730	0.56	0.41	14	65	2526
	7_S	0.07	0.31	7	68	2170	0.68	0.48	9	58	2526
	8_S	0.40	0.17	12	70	1870	0.50	0.38	18	50	2560

Table 4: Results of the developed classification models on the multi-labeled contributions

In multi-label classification tasks, the outputs of models are usually referred to as logits which are raw values from the final layer of a neural network. Logits are converted into probabilities or binary decisions according to a threshold. Determination of the most appropriate threshold for every class will be critical since class predictions depend directly on it. Thus, in the following part, the Area Under the Curve (AUC) is used to find the optimal threshold and is computed for every training fold. The optimum solution is the threshold that yields the best performance over the folds. The results after detecting and applying the best thresholds are given in Table 4.

As the results show, macro F-scores in the majority of the models have improved while micro F-scores have declined. This outcome is an indicator of the models' better performance in the

classification of rare classes. This is an expected result in the classification of unbalanced datasets. Moreover, contrary to the developed models on single-labeled texts, the base models have shown slightly better performance over multi-labeled data. This slightly lower performance of shortened texts is insignificant when compared to the reduction in train time. For Bonn, the variation in the F-scores is less significant. However, for Ehrenfeld and Moers, due to the presence of very rare classes, the combination of weighting and AUC has shown a significant improvement in both the micro and macro F-scores. More detailed information on the class-wise performance of the models can be found in Appendix 1.

6 Discussion

Based on the discussions in the previous section, we observed that the maximum length in the BERT models is an effective parameter. However, to provide the practitioners with a feasible argument data analytic system, limitations such as the ability to handle long texts, the requirement for less powerful systems, and the need for faster data analysis need to be addressed. Summarization is an effective technique for shortening the contributions and reducing the processing time while saving the key information. Based on the run-time values in Table 3, summarization reduces the train time using CPU by 50% on average without significantly affecting the performance. Reduction in time eases the use of models on systems without special specifications such as having a GPU. Moreover, the results from the trained models on the data of the city of Bonn showed that if a sufficient number of contributions is available under each label, truncation as a shortening technique performs close to the summarization technique. Thus, despite the removed parts of the texts, the model still has enough training instances to learn the distinguishing features of each class. On the contrary, if the number of contributions is limited, especially under each category, like in the case of Ehrenfeld, summarization preserves the text's key features and hence surpasses the performance of truncation in classification.

Our findings confirmed that similar conclusions are valid in the classification of multi-labeled contributions. However, like other multi-label classification problems, deciding on the optimal value of the thresholds plays a significant role in the performance of the models.

This study also compared the performance of two text summarization models for the German language on argument data which was left undiscovered. The summarized texts using the multilingual mBART model achieved around 2% higher performance in text classification against the BERT2BERT model which is specially fine-tuned for summarization of the German language. One of the potential reasons for the lower performance of the BERT2BERT model is the shorter length of the summarized documents. Shorter texts mean more loss of information which could be informative for the model. Although the quality of the summarized argument texts is not measured in this study, the results follow the findings of (Alcantara et al., 2023) on Wikipedia articles which confirmed that the mBART model outperforms the BERT2BERT model in abstractive German text summarization.

7 Conclusion and Future Work

The objective of this study was to explore the effect of abstractive summarization models in analyzing the German language argument data. Limited studies have worked on this topic and most of the existing ones are focused on the English language. Hence, this study was an attempt to fill this gap.

For this purpose, a BERT model fine-tuned in the German language (German BERT2BERT) and a BART model were considered. The German BERT2BERT fine-tuned on the MLSUM DE dataset was a reasonable candidate as it was specifically developed for German text summarization. The performance of the generated summarized texts using this model in the classification task was studied against the summarized texts from the BART model. Pre-trained on a large number of texts from different languages made the BART model a good alternative to the German BERT2BERT model.

The results proved that text summarization methods reduce the run-time of the model while keeping the classification quality almost the same. Additionally, our results suggest that the generated summarized texts using the German BERT2BERT model are briefer than the ones generated from the mBART model. Hence, the quality of the generated summarization is still open to be discovered. Moreover, in following the long-term goal of constructing a system for public contributions analysis, the need for developing a pre-trained model on the argument data is important.

Limitations

One of the obstacles in the analysis of German texts despite being a high-resource language is the shortage of high-quality labeled data. This limitation can either result in lower accuracy of the trained models or over-fitting due to the fine-tuning of the model on a specific in-reach dataset. Regarding the analysis of argument data, only a few labeled datasets exist. In this study, one of the important steps is the evaluation of the generated summarized texts. The first German Text Summarization Challenge was held in 2019 at the SwissText conference in which 100,000 texts collected from Wikipedia along with their reference summaries were analyzed (SwissText, 2019). However, to the best of our knowledge, this is the only available and suitable collection for the evaluation of summarization systems, in which each data is equipped with a referenced summary. Hence, for the summarization of argument texts, there is a lack of a developed dataset to evaluate the quality of models and generated summaries. This limitation also gets highlighted as there is a lack of benchmark studies in the summarization of German argument data. Finally, the size of available datasets is limited. The used dataset in this experiment consisted of 3139 contributions, while for some of the classes, the number of contributions is very low. A limited number of samples in a category compared to the other classes not only raises the challenges of unbalanced datasets but also negatively affects the model performance due to ineffective training.

References

Rowe, G., and Frewer, L. 2004. Evaluating Public Participation Exercises: A Research Agenda. *Science, Technology, and Human Values*. 29, 4. https://doi.org/10.1177/0162243903259197.

Romberg, J., and Escher, T. 2023. Making Sense of Citizens' Input through Artificial Intelligence: A Review of Methods for Computational Text Analysis to Support the Evaluation of Contributions in Public Participation. *Digital Government: Research and Practice*. 5. https://doi.org/10.1145/3603254.

Simonofski, A., Fink, J., and Burnay, C. 2021. Supporting policy-making with social media and e-participation platforms data: A policy analytics framework. *Government Information Quarterly*. 38, 3, 101590.

Lee, J., and Kim, S. 2014. Active Citizen E-Participation in Local Governance: Do Individual Social Capital and E-Participation Management Matter? *Proceedings of the Annual Hawaii International Conference on System Sciences*. 2044-2053. https://doi.org/10.1109/HICSS.2014.259.

Arana-Catania, M., van Lier, F.-A., Procter, R., Tkachenko, N., Zubiaga, A., and Liakata, M. 2021. Citizen Participation and Machine Learning for a Better Democracy. *Digital Government: Research and Practice*. 2. https://doi.org/10.1145/3452118.

Romberg, J., Mark, L., and Escher, T. 2022. A Corpus of German Citizen Contributions in Mobility Planning: Supporting Evaluation Through Multidimensional Classification. *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 2874–2883. Marseille, France.

Romberg, J., and Escher, T. 2022. Automated Topic Categorization of Citizens' Contributions: Reducing Manual Labelling Efforts Through Active Learning. In *Proceedings of the Conference*, 369–385. https://doi.org/10.1007/978-3-031-15086-9_24.

Egger, R., and Gokce, E. 2022. Natural Language Processing (NLP): An Introduction: Making Sense of Textual Data. https://doi.org/10.1007/978-3-030-88389-8_15.

Romberg, J. 2023. Mind the User! Measures to More Accurately Evaluate the Practical Value of Active Learning Strategies. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, 996–1006. https://aclanthology.org/2023.ranlp-1.107.

Balta, D., Kuhn, P., Sellami, M., Kulus, D., Lieven, C., and Krcmar, H. 2019. How to Streamline AI Application in Government? A Case Study on Citizen Participation in Germany. In *Lindgren, I., et al. (Eds.), Electronic Government. EGOV 2019*, Lecture Notes in Computer Science, vol. 11685. Springer, Cham. https://doi.org/10.1007/978-3-030-27325-5_18.

Romberg, J., and Conrad, S. 2021. Citizen Involvement in Urban Planning - How Can Municipalities Be Supported in Evaluating Public Participation Processes for Mobility Transitions? In *Proceedings of the 8th Workshop on Argument Mining*, ACL, Punta Cana, 88–99. https://aclanthology.org/2021.argmining-1.9.

Romberg, J. 2022. Is Your Perspective Also My Perspective? Enriching Prediction with Subjectivity. In *Proceedings of the 9th Workshop on Argument Mining*, International Conference on Computational Linguistics, Gyeongju, Republic of Korea, 115–125. https://aclanthology.org/2022.argmining-1.11.

Aruneshwari, R. R., Anandkumar, K. M., and Kavitha, D. 2024. A Comprehensive Review of Text

Summarization. *AIP Conference Proceedings*. 2802, 1, 140003. https://doi.org/10.1063/5.0186988.

Alcantara, T. H. M., Krütli, D., Ravada, R., and Hanne, T. 2023. Multilingual Text Summarization for German Texts Using Transformer Models. *Information*. 14, 6, 303. https://doi.org/10.3390/info14060303.

Verma, P., and Verma, A. 2020. A Review on Text Summarization Techniques. *Journal of Scientific Research*. 64, 251–257. https://doi.org/10.37398/JSR.2020.640148.

Talib, R. 2021. Multilingual Text Summarization using Deep Learning. *International Journal of Engineering Research and Advanced Technology*. 7, 29–39.https://doi.org/10.31695/IJERAT.2021.3712.

Anguiano, E., Villaseñor-Pineda, L., Montes, M., and Rosso, P. 2010. Summarization as Feature Selection for Document Categorization on Small Datasets. In *Proceedings of the Conference*, 39–44. https://doi.org/10.1007/978-3-642-14770-8_6.

Dewi, K., and Sagala, R. 2018. Using Summarization to Optimize Text Classification. *IOP Conference Series: Materials Science and Engineering*. 407, 012157. https://doi.org/10.1088/1757-899X/407/1/012157.

Khan, B., Shah, Z., Usman, M., Khan, I., and Niazi, B. 2023. Exploring the Landscape of Automatic Text Summarization: A Comprehensive Survey. *IEEE Access*. PP, 1–1. https://doi.org/10.1109/ACCESS.2023.3322188.

Scialom, T., Dray, P.-A., Lamprier, S., Piwowarski, B., and Staiano, J. 2020. MLSUM: The Multilingual Summarization Corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 8051–8067. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.647.

Schroeter, R., Scheel, O., Renn, O., and Schweizer, P.-J. 2016. Testing the Value of Public Participation in Germany: Theory, Operationalization and a Case Study on the Evaluation of Participation. *Energy Research & Social Science*. 13. https://doi.org/10.1016/j.erss.2015.12.013.

Karic, S., Althaus, M.-C., and Heissler, J. 2024. Digital and Multi-Channel Citizen Participation in Germany: A Comprehensive Overview of Patterns, Methods and Determinants. *Raumforschung und Raumordnung | Spatial Research and Planning*. 82. https://doi.org/10.14512/rur.2170.

Kwon, N., Shulman, S., and Hovy, E. 2006. Multidimensional Text Analysis for eRulemaking. In *ACM International Conference Proceeding Series*, 151, 157–166. https://doi.org/10.1145/1146598.1146649.

Arguello, J., Callan, J., and Shulman, S. 2008. Recognizing Citations in Public Comments. *Journal of Information Technology & Politics*. 5, 49–71. https://doi.org/10.1080/19331680802153683.

Habernal, I., and Gurevych, I. 2016. Which Argument is More Convincing? Analyzing and Predicting Convincingness of Web Arguments Using Bidirectional LSTM. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, 1589–1599. Berlin, Germany. https://doi.org/10.18653/v1/P16-1150.

Cardie, C., Farina, C., Aijaz, A., Rawding, M., and Purpura, S. 2008. A Study in Rule-Specific Issue Categorization for E-Rulemaking. In *Proceedings of the Conference*, 244–253. https://doi.org/10.1145/1367832.1367874.

Konat, B., Lawrence, J., Park, J., Budzynska, K., and Reed, C. 2016. A Corpus of Argument Networks: Using Graph Properties to Analyse Divisive Issues. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 3899–3906. https://aclanthology.org/L16-1617.

Fierro, C., Pérez, J., Quezada, M., and Fuentes Bravo, C. 2017. 200K+ Crowdsourced Political Arguments for a New Chilean Constitution. https://doi.org/10.18653/v1/W17-5101.

Liebeck, M., Esau, K., and Conrad, S. 2016. What to Do with an Airport? Mining Arguments in the German Online Participation Project Tempelhofer Feld. In *Proceedings of the Workshop on Argument Mining*, 144–153. https://doi.org/10.18653/v1/W16-2817.

Ker, S. J., and Chen, J. N. 2000. A Text Categorization Based on Summarization Technique. In *Proceedings of the ACL-2000 Workshop on Recent Advances in Natural Language Processing and Information Retrieval: Held in Conjunction with the 38th Annual Meeting of the Association for Computational Linguistics - Volume 11 (RANLPIR '00)*, 79–83. Association for Computational Linguistics, USA. https://doi.org/10.3115/1117755.1117766.

Zhao, J., and Gui, X. 2017. Comparison Research on Text Pre-processing Methods on Twitter Sentiment Analysis. *IEEE Access*, PP. 1-1. https://doi.org/10.1109/ACCESS.2017.2672677.

Hovy, E., and Lin, C. Y. 1998. Automated Text Summarization and the Summarist System. In *Proceedings of a Workshop*, 197–214. https://doi.org/10.3115/1119089.1119121.

Syed, A., Gaol, F., and Matsuo, T. 2021. A Survey of the State-of-the-Art Models in Neural Abstractive Text Summarization. *IEEE Access*, 9, 13248–13265. https://doi.org/10.1109/ACCESS.2021.3052783.

Chen, C., Yin, Y., Shang, L., Jiang, X., Qin, Y., Wang, F., Wang, Z., Chen, X., Liu, Z., and Liu, Q. 2021. bert2BERT: Towards Reusable Pretrained Language Models. *arXiv preprint*. https://arxiv.org/abs/2110.07143.

GitHub repository. 2022. Available online: https://github.com/google-research/bert/. Accessed on 9 September 2024.

Scialom, Thomas, Dray, Paul-Alexis, Lamprier, Sylvain, Piwowarski, Benjamin, & Staiano, Jacopo. (2020). MLSUM: The Multilingual Summarization Corpus. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 8051–8067). https://doi.org/10.18653/v1/2020.emnlp-main.647.

Cukier, R. (2024). Increasing transformer token length with a Maximum Entropy Principle Method. *arXiv preprint arXiv:2408.10277*. https://doi.org/10.48550/arXiv.2408.10277.

Hugging Face. (2024). "Chapter 2: Transformer Models" *NLP Course*. Accessed September 10, 2024. https://huggingface.co/learn/nlp-course/chapter2/5?fw=tf.

Chi, Sun & Qiu, Xipeng & Xu, Yige & Huang, Xuanjing. (2019). How to Fine-Tune BERT for Text Classification?.*arXiv*.https://doi.org/10.48550/arXiv.1905.05583.

Mutasodirin, Mirza & Prasojo, Radityo. (2021). Investigating Text Shortening Strategy in BERT: Truncation vs Summarization. *Proceedings of the International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 1-5. https://doi.org/10.1109/ICACSIS53237.2021.9631364.

Lewis, Mike & Liu, Yinhan & Goyal, Naman & Ghazvininejad, Marjan & Mohamed, Abdelrahman & Levy, Omer & Stoyanov, Veselin & Zettlemoyer, Luke. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL 2020)*, 7871-7880. https://doi.org/10.18653/v1/2020.acl-main.703.

Ma, Edward. (2019). NLP Augmentation. *GitHub repository*. https://github.com/makcedward/nlpaug.

Wolf, Thomas & Debut, Lysandre & Sanh, Victor & Chaumond, Julien & Delangue, Clement & Moi, Anthony & Cistac, Pierric & Rault, Tim & Louf, Remi & Funtowicz, Morgan & Davison, Joe & Shleifer, Sam & Platen, Patrick & Ma, Clara & Jernite, Yacine & Plu, Julien & Xu, Canwen & Scao, Teven & Gugger, Sylvain & Rush, Alexander. (2020). Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 38-45. https://doi.org/10.18653/v1/2020.emnlp-demos.6.

SwissText. (2019). German Text Summarization Challenge. Retrieved from https://www.swisstext.org/2019/shared-task/german-text-summarization-challenge.html

A Additional Literature Review

NLP in Public Contributions Over the past 50 years, there have been growing discussions regarding the public's role in determining policies. However, the majority of the existing research is limited to some basic analyses (Simonofski, Fink, and Burnay, 2021). “Citizen Participation and Machine Learning for a Better Democracy” is one of the recent studies that highlight how effective NLP is in getting over the difficulties of analyzing public participation platforms (Arana-Catania et al., 2021). Automatic Text Summarization is one of the branches of NLP and its importance has been highlighted in recent years due to the advances in technology.

Text Summarization The Vast generation of information in the form of documents requires a searching system to access usable information efficiently, save time, and convey the main and important information to the reader (Alcantara et al., 2023). Text summarization is one of the solutions. Early experiments applied the traditional text summarization technique such as weighting terms according to their frequencies (Ker & Chen, 2000), as a feature selection strategy to improve classification performance (Anguiano et al., 2010, Dewi and Sagala, 2018). (Zhao & Gui, 2017) emphasized the preprocessing stage as a method of summarization. These techniques, however, were mostly based on the naïve properties of the texts such as sentence position and word frequency counts (Khan et al., 2023). With the advances in neural networks and deep learning, techniques focused more on abstractive summarization methods using pre-trained models (Khan et al., 2023). The work of Hovy and Lin (1998) is one the first ones in Automatic Text Summarization (ATS) (Hovy & Lin, 1998). They developed the SUMMARIST system which benefited from

language-specific techniques of parsing and semantic analysis combined with Information Retrieval (IR) and statistical methods.

B Dataset Distribution

In deciding on the best document length, Figure 1 provides a clearer depiction of the contributions length distribution. It can be observed that the distributions are skewed to the left and the length of the most of contributions is less than 128 tokens.

Figure 2 shows the distribution of the summarized texts using the mBART model. It can be seen that the lengths are more evenly distributed.

Figure 3 illustrates the texts' length distribution after summarization using the BERT2BERT model. In comparison with the mBART model, the lengths are more skewed to the left. However, a similar pattern is observed in the summarized texts' distribution.

C Class-wise performance of the multi-labeled models

Table 5 provides detailed information on the performance of the trained models on the multi-labeled contributions. Table 6 shows the results after detecting and applying the best thresholds. As it can be observed, using AUC and finding the optimal threshold has significantly improved the performance of models, especially in classifying contributions of less frequent classes. In Table 6 For Ehrenfeld, except for the lighting category, we have an improvement in the F-scores, especially when the weighting is applied.

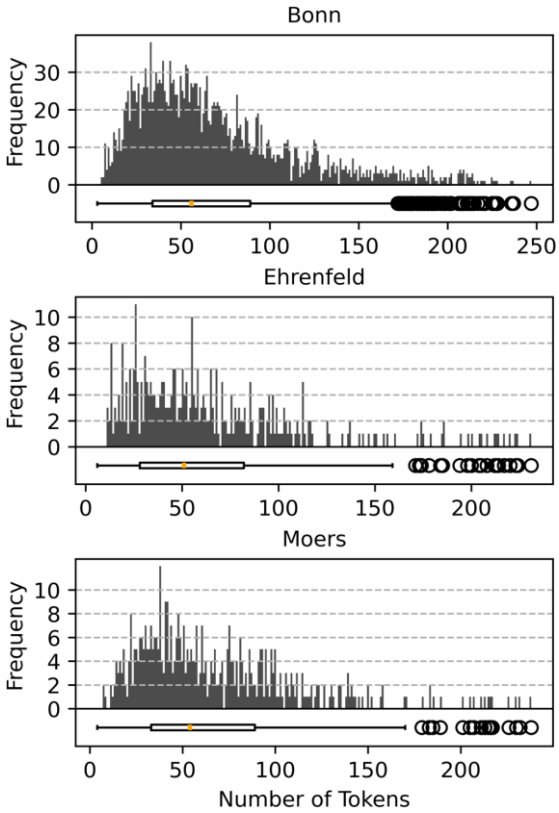

Figure 1: Distribution of contributions length

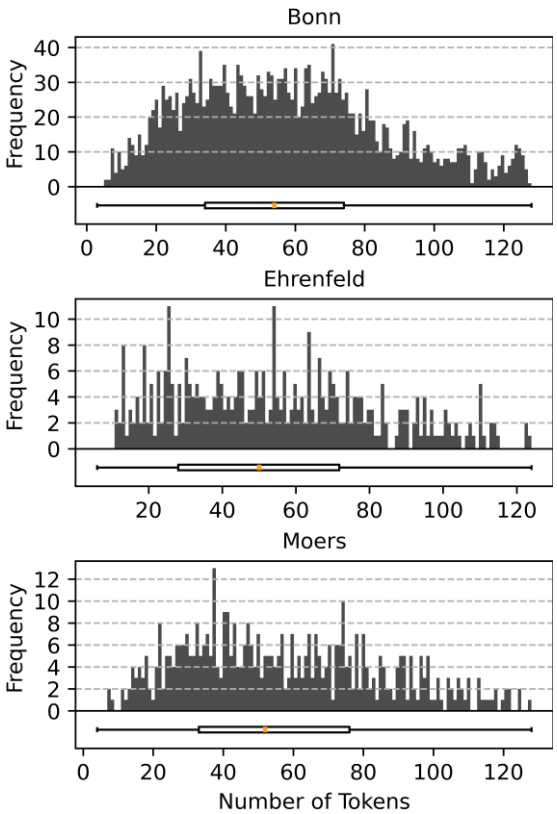

Figure 2: Distribution of the length of the contributions after summarizing texts using the multilingual mBART model

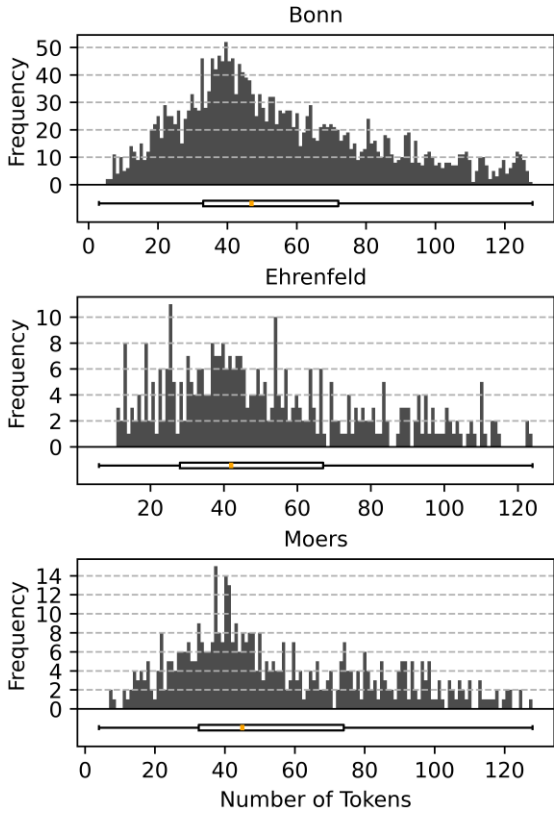

Figure 3: Distribution of the length of the contributions after summarizing texts using the BERT2BERT model

	Experiments	Bicycle Parking	Cycle Path Quality	Cycling Traffic Management	Lighting	Misc	Obstacles	Signage	Traffic Lights	Micro F-score	Macro F-score	Hamming	Train run-time (second) GPU	Train run-time (second) CPU
Bonn	**1_B**	0.921	0.777	0.809	0.833	0.044	0.769	0.56	0.824	0.774	0.692	0.059	383.8	17482
	2_B	0.919	0.739	0.798	0.833	0.329	0.663	0.596	0.820	0.753	0.712	0.063	383.2	18378
	3_T	0.909	0.771	0.814	0.804	0.084	0.767	0.571	0.821	0.774	0.693	0.059	218.8	9955
	4_T	0.923	0.729	0.795	0.856	0.332	0.665	0.588	0.809	0.750	0.712	0.064	221.0	8940
	5_S	0.904	0.769	0.816	0.829	0.063	0.765	0.553	0.817	0.773	0.689	0.059	219.8	9935
	6_S	0.909	0.728	0.794	0.862	0.341	0.665	0.563	0.803	0.747	0.708	0.065	224.6	9702
	7_S	0.898	0.768	0.799	0.800	0.043	0.746	0.515	0.806	0.759	0.672	0.063	233.8	9120
	8_S	0.898	0.717	0.784	0.839	0.270	0.642	0.540	0.805	0.735	0.687	0.067	233.0	10128
Ehrenfeld	**1_B**	0.762	0.631	0.854	0	0	0.573	0.194	0.657	0.722	0.459	0.073	93.8	2782
	2_B	0.561	0	0.751	0	0	0	0.279	0.295	0.571	0.236	0.116	93.2	2930
	3_T	0.779	0.629	0.859	0	0	0.568	0.313	0.697	0.734	0.481	0.070	62.6	1635
	4_T	0.364	0	0.740	0	0	0	0.08	0.239	0.546	0.177	0.121	65.2	1669
	5_S	0.780	0.632	0.855	0	0	0.542	0.247	0.685	0.727	0.467	0.072	47.2	1750
	6_S	0.466	0	0.735	0	0	0	0.08	0.222	0.545	0.188	0.121	45.6	1534
	7_S	0.669	0.546	0.847	0	0	0.512	0.293	0.666	0.701	0.441	0.079	47.6	1517
	8_S	0.350	0	0.736	0	0	0	0.160	0.233	0.545	0.184	0.121	45.6	1619
Moers	**1_B**	0	0.810	0.813	0	0	0	0	0.829	0.713	0.306	0.068	85.2	4406
	2_B	0.533	0.062	0.559	0	0	0	0	0.250	0.388	0.175	0.123	85.4	3730
	3_T	0	0.828	0.831	0	0	0	0	0.832	0.727	0.311	0.066	56.8	1987
	4_T	0.533	0.048	0.582	0	0	0	0	0.258	0.404	0.177	0.121	76.8	1837
	5_S	0	0.799	0.815	0	0	0	0	0.861	0.716	0.309	0.069	53.8	2099
	6_S	0.533	0.016	0.586	0	0	0	0	0.240	0.398	0.172	0.123	54.8	1730
	7_S	0	0.796	0.806	0	0	0	0	0.849	0.071	0.306	0.070	67.6	2170
	8_S	0.533	0.016	0.583	0	0	0	0	0.258	0.399	0.173	0.121	70.0	1870

Table 5: Performance of the developed classification models on the multi-labeled contributions.

	Experiments	Bicycle Parking	Cycle Path Quality	Cycling Traffic Management	Lighting	Misc	Obstacles	Signage	Traffic Lights	Micro F-score	Macro F-score	Hamming	Train run-time (second) GPU	Train run-time (second) CPU
Bonn	**1_B**	0.901	0.779	0.825	0.855	0.349	0.733	0.607	0.780	0.761	0.729	0.071	430.0	22606
	2_B	0.908	0.741	0.808	0.723	0.506	0.712	0.599	0.799	0.754	0.725	0.071	426.8	22126
	3_T	0.913	0.778	0.823	0.879	0.383	0.718	0.581	0.754	0.760	0.729	0.072	336.0	11529
	4_T	0.920	0.740	0.799	0.738	0.464	0.707	0.622	0.806	0.749	0.724	0.073	301.4	12300
	5_S	0.917	0.766	0.818	0.829	0.354	0.725	0.550	0.780	0.750	0.717	0.075	256.2	13814
	6_S	0.908	0.735	0.794	0.586	0.436	0.715	0.568	0.804	0.741	0.693	0.076	269.4	12743
	7_S	0.900	0.762	0.812	0.856	0.306	0.710	0.537	0.777	0.739	0.708	0.080	329.4	13791
	8_S	0.902	0.730	0.790	0.783	0.419	0.689	0.567	0.762	0.732	0.705	0.078	251.6	12164
Ehrenfeld	**1_B**	0.654	0.537	0.814	0	0.363	0.286	0.391	0.442	0.633	0.436	0.106	71.0	3877
	2_B	0.426	0.319	0.641	0	0.449	0.36	0.094	0.148	0.403	0.304	0.279	71.4	4100
	3_T	0.599	0.573	0.816	0	0.387	0.38	0.316	0.442	0.642	0.439	0.100	45.0	2219
	4_T	0.463	0.266	0.552	0	0.398	0.338	0.219	0.153	0.362	0.299	0.291	47.6	2290
	5_S	0.495	0.554	0.822	0	0.277	0.287	0.358	0.453	0.633	0.406	0.107	44.4	1801
	6_S	0.355	0.246	0.626	0	0.59	0.306	0.199	0.116	0.398	0.305	0.257	46.4	1765
	7_S	0.483	0.492	0.796	0	0.345	0.445	0.363	0.552	0.628	0.431	0.104	49.6	2056
	8_S	0.473	0.307	0.646	0	0.44	0.299	0.147	0.290	0.409	0.325	0.283	52.0	1985
Moers	**1_B**	0.333	0.790	0.820	0.406	0.260	0.044	0.325	0.829	0.670	0.476	0.093	84.8	5449
	2_B	0.533	0.469	0.699	0.180	0.230	0.165	0.217	0.597	0.504	0.386	0.165	87.2	4820
	3_T	0.223	0.804	0.823	0.311	0.249	0.292	0.345	0.856	0.670	0.488	0.095	54.2	2800
	4_T	0.333	0.612	0.636	0.179	0.179	0.190	0.337	0.875	0.468	0.418	0.197	54.6	2605
	5_S	0.466	0.822	0.837	0.381	0.252	0.160	0.406	0.822	0.663	0.518	0.100	59.8	2770
	6_S	0.333	0.488	0.708	0.196	0.338	0.140	0.233	0.820	0.555	0.407	0.142	65.4	2526
	7_S	0.333	0.809	0.834	0.530	0.138	0.121	0.363	0.737	0.682	0.483	0.087	58.4	2526
	8_S	0.333	0.490	0.683	0.215	0.298	0.134	0.243	0.627	0.495	0.378	0.183	50.4	2560

Table 6: Performance of the developed classification models on the multi-labeled contributions. Thresholds are optimized using AUC.