ORIGINAL ARTICLE

Data augmentation in natural language processing: a novel text generation approach for long and short text classifiers

Markus Bayer[1] · Marc-André Kaufhold[1] · Björn Buchhold[2] · Marcel Keller[3] · Jörg Dallmeyer[2] · Christian Reuter[1]

Abstract

In many cases of machine learning, research suggests that the development of training data might have a higher relevance than the choice and modelling of classifiers themselves. Thus, data augmentation methods have been developed to improve classifiers by artificially created training data. In NLP, there is the challenge of establishing universal rules for text transformations which provide new linguistic patterns. In this paper, we present and evaluate a text generation method suitable to increase the performance of classifiers for long and short texts. We achieved promising improvements when evaluating short as well as long text tasks with the enhancement by our text generation method. Especially with regard to small data analytics, additive accuracy gains of up to 15.53% and 3.56% are achieved within a constructed low data regime, compared to the no augmentation baseline and another data augmentation technique. As the current track of these constructed regimes is not universally applicable, we also show major improvements in several real world low data tasks (up to +4.84 F1-score). Since we are evaluating the method from many perspectives (in total 11 datasets), we also observe situations where the method might not be suitable. We discuss implications and patterns for the successful application of our approach on different types of datasets.

Keywords Textual data augmentation · Small text data analytics · Text generation · Long and short text classifier

1 Introduction

Deep learning has attracted considerable attention due to increased computing power in combination with a higher availability of training data for a wide range of problems [55]. In some learning tasks, especially small data regimes, the development of training data might have a higher relevance than the choice and modelling of classifiers [4]. To improve classifiers, data augmentation methods have been designed to artificially create training data with specific transformations [56]. Current research in data augmentation focuses on deep learning algorithms, which are state of the art for many classification tasks, as they still often suffer from a strong variance regarding the given problem if not enough data is provided. The artificial creation of training data serves as a kind of regularization and thus, simpler solutions are preferred [14, 63]. In addition, imbalance in datasets can be addressed [40, 64] and the security of classifiers can be increased by making them resistant to the deception by skillful changes in the input sequences [35]. Data augmentation can also help to mitigate the "big data wall" problem, which relates to the fact that smaller companies, research groups and organizations are usually unable to acquire the same volume of data as large corporations [10].

Regardless of deep learning, research into artificial data creation can benefit natural language processing (NLP) applications across several domains where training data is scarce or labelling is costly. For example, to enhance the situational awareness of emergency managers, a part of crisis informatics deals with the rapid recognition and subsequent classification of messages and pictures during disasters and emergencies [1, 21]. Due to the scarcity of financial and personnel resources, emergency services lose valuable

✉ Markus Bayer
bayer@peasec.tu-darmstadt.de

Marc-André Kaufhold
kaufhold@peasec.tu-darmstadt.de

Christian Reuter
reuter@peasec.tu-darmstadt.de

1 Technical University of Darmstadt, Darmstadt, Germany

2 CID GmbH, Freigericht, Germany

3 Hanau, Germany

time dealing with complicated identification tasks, which eventually can cost lives [17, 43]. This problem of scarcity also applies to small- and medium-sized enterprises (SMEs) when requiring a high quality and volume of labelled data for commercial tasks such as brand analysis or news classification [53]. In NLP, there is the difficulty establishing universal rules for transformations of textual data that can be carried out automatically and still maintain the quality of the labelling, which is especially sensitive in domains such as sentiment analysis [33]. Ref. [32] suggest that current pre-training or transfer learning methods in NLP already cover the goals of data augmentation. They argue that augmentation methods that only perturb the input data and do not provide new linguistic patterns fail to increase the classification quality of pre-trained models.

Thus, we propose a sophisticated generation-based method that overcomes these problems by incorporating new linguistic patterns (i.e., a high grammatical variety) which prove to be useful in combination with pre-trained models. This method does not simply create very similar instances, but very novel ones. Our approach uses two sub methods, whereof one is context-conditional by incorporating parts of the instances (e.g., first words or title) in the generation process and hence suited for long texts, while the other is context independent and suited for short texts. Although there is no clear distinction between long and short texts, we are guided by the 280 character limit (i.e., the length of a message in Twitter), at which most standard NLP data sets would be categorized as small. Thus, we seek to answer three research questions: How can we utilize text generation approaches of data augmentation that achieve a high novelty in the data while preserving the label quality to improve pre-trained machine learning classifiers (**RQ1**)? In which way is the incorporation of contexts of long text instances in classification problems helpful when using text generation as data augmentation method (**RQ1.1**)? How is it possible to achieve a quality improvement for classification tasks with short texts when augmenting with text generation (**RQ1.2**)?

Contributing to the domain of small data analytics, our results indicate additive accuracy gains of up to 15.53% and 3.56% within a constructed low data regime, compared to the no augmentation baseline and another data augmentation technique. As the current track of these constructed regimes is not universally applicable, we also show major improvements in several real world low data tasks (up to +4.84 F1-score). Since we are evaluating the method from many perspectives (in total 11 datasets), we also observe situations where the method might not be suitable. We discuss empirical (i.e., insights into the domain-specific application of small data analytics), practical (i.e., new data augmentation methods based on the GPT-2 language model) and theoretical (i.e., a textual data augmentation basis which is beneficial for pre-trained classification models) implications

for the successful application of our approach on different types of datasets.

The paper is structured as follows: After introducing related work on data augmentation, NLP and text generation approaches (Sect. 2), the paper presents both the concept and implementation of a novel text generation data augmentation algorithm (Sect. 3). Furthermore, it presents the method and findings of three rounds of evaluation (Sect. 4) before discussing the implications, limitations and potentials for future research (Sect. 5).

2 Related work

2.1 Foundations of data augmentation

Data augmentation is a machine learning technique that artificially enlarges the amount of training data by means of label preserving transformations [56]. First variations of data augmentation can be identified in the well-known LeNet by [30]. Using random distortions of training pictures, the MNIST-dataset was ninefold enlarged, so that a better detection of handwritten digits became feasible. A relevant term of data augmentation is label preservation, describing transformations of training data that preserve class information [10]. This means that this kind of transformations modifies texts of a given class to other texts that are as well related to this class. In data augmentation research, this is of high relevance because the absence of it would result in the generation of incorrectly classified data. For the most part, an entity replacement within a sentence is sufficient for label preservation in sentiment analysis. However, the random addition of words may result in an alteration of the sentiment. Many researchers loosen the label preservation term. Then, transformations that break the preservation are legitimate as long as the label is adjusted simultaneously. Furthermore, transformations that preserve the right class with a high probability, but not with certainty, may exist. In this understanding, [48] designate the probability that the correct label is assigned after a transformation as the safety of a data augmentation method. For example, this uncertainty, if known, could be directly integrated in the label. If unknown, methods like label smoothing can model a general uncertainty.

In NLP, data augmentation is considered a difficult task [19] since textual transformations that preserve the label are difficult to define [24, 59]. Thus many methods have been tried out in research so far. Among them are methods for swapping [59], deleting [16, 38], inducing spelling mistakes [6, 10], paraphrasing [28], and replacing of synonyms [25, 61, 66], close embeddings [2, 58] and words predicted by a language model [11, 18, 24] on word-level. On a broader level, methods which change the dependency tree [45, 62],

perform round-trip-translation [27, 47], or interpolate the input instances [9, 65] are used. Further studies have dealt with text generation approaches for data augmentation. While [44] and [54] are using recurrent neural networks and generative adversarial networks for short-text augmentation, [38] sample instances from a variational autoencoder without length restrictions. Furthermore, [57] and [3] used the GPT-2 model for text generation. A more detailed analysis, taxonomy and listing of data augmentation techniques can be found in the data augmentation survey by [5].

However, challenging many research directions in this area, [32] hypothesize that textual data augmentation would only be helpful if the generated data contains new linguistic patterns that are relevant to the task and have not yet been seen in pre-training.

2.2 Research gap

Our data augmentation method is inspired by the text generation methods from [38, 44, 54] while also considering the limitations outlined by [32] and seeks to tackle three primary research gaps:

1. Considering short and long texts while maintaining coherence and achieving high novelty;
2. preserving the labels and quality of the augmentation method;
3. overcoming the challenge of limited usefulness of textual data augmentation in combination with pre-trained models.

First, in contrast to the works of [44] and [38], we consider short as well as long texts as input data instances to our augmentation method, which is covered explicitly by research questions 1.1 and 1.2. Additionally, and in relation to the main research question, our method is characterized by substantial label preservation in combination with the novelty and coherence of the data. At first, the generation is enriched with a special finetuning and prefix addition. Then, a document embedding filter is applied so that instances not associated with the actual class are omitted. Thus, the generation capabilities can be used in full extent while tailoring them to the class data. Furthermore, our experiments are based on the GPT-2-Model by [39] that achieves very good results in text generation.

Second, when it comes to usage of the GPT-2 model, [57] describe no measures for label and quality preservation in their GPT-2 augmentation. Ref. [3] indicate that the GPT-2 model will be further trained and improperly generated instances will be removed. In contrast to our method, the model of [3] is limited exclusively on sentences as instances and cannot generate coherent text. Furthermore, it uses other safety mechanisms for label preservation. For instance, they use a filter mechanism based on a classifier, that was trained on the class data. This can severely reduce the diversity of the data augmentation method.

Third, the method proposed in this paper is intended to overcome the issue that textual data augmentation can be of no or small value when used in combination with pre-trained classifiers [32]. In contrast to the study by [32], we use the ULMFit model by [13]. Nevertheless, the model is also pre-trained beforehand and finetuned on each task dataset. As a specialty, we also finetune the encoder with the augmented data for the baseline, making sure that at least the encoder has seen all linguistic patterns before.

3 Concept and implementation

3.1 Conceptual design

The text generation process can be based on any language model with good text generation capabilities. Language models indicate a probability distribution of sequences of words:

$$P_{\Theta}(w_t | w_{t-k}, \dots, w_{t-1}) \, \forall t \tag{1}$$

The model P_{Θ} predicts the probability that the current word is w_t given the predecessor words (context) $w_{t-1}, \dots, w_{t-k}$. This enables the P_{Θ} to generate texts. A phrase prefix can be used as context to make the model follow a certain topic by completing exactly this part, abstracting from exact specifics as sampling methods. In addition, a temperature parameter can be introduced to adjust the randomness in the generation of the texts by scaling the logits in the softmax. In order to enable the sensible use of a language model for data augmentation, it has to be ensured that the procedure mainly generates texts which are similar to the training data and, in addition, reflect the respective class (label preservation or safety). In the following, our augmentation method is described, which comprises the specification of three steps for modeling this behavior.

In a first step, the pre-trained model P_{Θ} is further trained (finetuning) with the training data X_c of the class c that should be enriched. On the one hand, this enables the model to learn the words, spelling and form of the training data. On the other hand, a bias is generated with regard to the selected class. This means that, for the generation of data, the selected class can be retained more explicitly. In the following, we differentiate between the contextual data augmentation process that is suitable for longer texts and a context independent process for shorter training instances.

In order to further strengthen the safety and label preservation, special "start of text" -tokens are added to each training data in the finetuning input. In the text generation phase,

these tokens are used as generation prefixes, signaling the model to generate texts similar to the specific training data. This ensures that the augmented examples are different to each other but remain based on the actual data. If the training data consists of longer texts, i.e. instances containing more than 280 characters, this token can be selected context-based, for example, by appending the first words or the title of each instance (e.g. "$< |startoftext| > (w_1 \ldots w_k)_i$" where $(w_1 \ldots w_k)_i$ is the beginning sequence of the instance i). This results in a high diversity of the generated data. Although, if the texts of the dataset are short and no context tokens can be used for appendage, the context independent variant is chosen, where the number of the occurrence of the instance in the training set is concatenated (e.g. "$< |startoftext| > |i|$" where i is the occurrence). As the language model is finetuned on the training data, it can be assumed that it learns to associate the unique token with the respective instance. Thereby, the model is able to recognize the prefix and completes it on the basis of memorization. Ultimately, this implies a strengthened label preservation. However, so that the data is not completely reproduced from memory, uncertainty is introduced in the sampling by adjusting the temperature parameter.

Filtering the generated data is the final heuristic to increase label preservation. For this purpose, document embeddings for each instance of the generated texts and training data of a class are created. The embeddings reflect the content of the respective instances. If in this latent space a data instance from the generated data X_{gen} is too far away from the actual training data X_c of the class to be augmented, it can be assumed that the content differs semantically and/or syntactically, which is why such data is discarded:

$$X_{filtered} = \{x_i \in X_{gen} \mid dist(Emb(x_i), Centroid(Emb(X_c))) < \delta\} \tag{2}$$

The large generative model is able to interpolate textual content in a sensible and non-trivial way. These capabilities are very promising for data augmentation by creating highly diverse samples that are coherent and contain new linguistic and semantic patterns with regards to the actual data. However, only through the application of the security steps can the model generate class-related data that does not represent the wrong label.

3.2 Implementation

Figure 1 shows and summarizes the three different steps of the safety enhancement, sorted according to algorithmic order. The class safety of the procedure can be significantly increased by this, although it cannot be completely ruled out that the correct label is obtained. For the implementation, we use GPT-2 by [39] with 355 million parameters. We used GPT-2 as it is well suited for small data
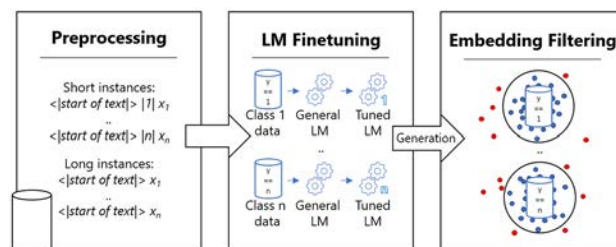

Fig. 1 Three steps to increase the probability of a label preserving instance in GPT-2 text generation (safety increase). In the first step, a contextualized or numbered start token is added to each training instance so that this can be used as a generation prefix for each class after the second step, in which the GPT-2 model is trained further. After the text generation, in a last step a filtration is carried out using BERT document embeddings so that significantly deviant instances are not included

analytics due to its diverse generation capabilities coming from the size. The model is enriched with the three different extensions discussed in the conceptual design section.

In the first steps, the GPT-2 model is imported and the specific class data is extracted. Subsequently, all instances of this class are given a prefix token ("$< |startoftext| > |\{num\}|$") and suffix token ("$< |endoftext| >$"). If the training data consists of longer instances with an embeddable context, the "$|\{num\}|$"-field is removed. In all other cases "$\{num\}$" is replaced by the position of the current data instance. Afterwards, the model is finetuned with this data several hundred or thousand epochs, dependent on the dataset size, so that the loss of the model is greatly reduced. This should sufficiently ensure that the model prioritizes the training data in the generation.

Thereafter, texts are generated for each class instance. If the training instances exceed a certain number of words they are considered as long and the token "$< |startoftext| >$" and a specific context of the document (e.g. title or the first words) are added to the generation, else "$< |startoftext| >$" is used in combination with the index of the respective instance. Temperatures between 0.7 and 0.9 should be set in the generation step [60], whereby a higher number represents greater randomness/creativity. In the last step of the procedure, the generated data is filtered. This is done by using Sentence-BERT [41] to create document embeddings of the data. Generated instances that are, according to a manually set threshold, too far away from the centroid of the correct data, are deleted from the result set. To minimize this interaction, a predefined value (e.g. 0.3) is set and the algorithm displays the 10 furthest instances that are still within this threshold. Depending on how many instances are wrong, the threshold is moved further and the process is started again. For example, if there is one false instance in the set of 10 generated instances, the threshold is increased slightly. If

there are only true instances, the threshold is decreased. This is done until a meaningful parameter is found.

The algorithms for long and short instances are given in Appendix C.

4 Evaluation

4.1 Selection of application domains

To conduct our evaluation, we selected the three cases of sentiment analysis, news classification and crisis informatics. First, sentiment analysis is, according to [33], the analysis of opinions, attitudes and emotions toward individuals, events or topics. It is a very common NLP task and used in a broad variety of applications, as, for example, the decision making process of organisations and individuals is increasingly dependent on public opinions [31]. As it is part of several textual machine learning benchmarks and highly used in research, it is a sensible task for the experiments of this paper.

To place this experiment in the setting of small data analytics, an artificially downsampled version will be used.

Second, due to the constantly growing number of news items and their information sources, it is becoming more and more complicated keeping track of topics and finding specific articles [8]. Classifiers are used to automatically divide news into predefined classes [26]. However, news are highly dynamic, so that the source domains are constantly shifting and new classes emerge. As an extreme example in this context, one can compare the news landscape before and after COVID-19 occurred. Such shifts and newly emerging topics have as result that new data has to be labeled all the time, leading to classifiers with small data bases. While this is already sufficient to form a focus in this study, we are also interested in exploring classification as well as data augmentation for large texts, as this receives little attention in research.

Third, the research field of crisis informatics draws on computing and social science perspectives to study the ways in which ICT enables, constrains, and mediates human practices related to crisis and disaster [51]. Besides the topics of crisis communication, community interaction, and inter-organizational collaboration [42], crisis informatics examines the application of machine learning to reduce the information overload of irrelevant information, extract useful information from social media (e.g., eyewitness reports, multimedia files), and enhance information quality for both an improved situational awareness and decision making of emergency services [20]. Despite the considerable volumes of social big data disseminated during large-scale emergencies, there is a class imbalance since only a small number of social media posts contribute to situational awareness [1]. Furthermore, emergency services such as fire or police departments often lack the financial and personnel resources to engage in comprehensive dataset labelling tasks [17, 43]. In contrast, there might be a lack of available raw data in small-scale and uncommon types of emergencies, qualifying crisis informatics as an interesting application field for data augmentation and small data analytics.

4.2 Model and datasets

In accordance with the research questions, the previously conceptualized and implemented data augmentation methods are evaluated in this chapter based on a constructed low data regime with the SST-2 dataset (Results I) and real world low data regimes regarding topic classification of long (Results II) and short texts (Results III). We used the ULMFit model by [13] that consists of a pre-trained encoder coupled with a linear pooling network and a softmax output. The encoder is finetuned for each task on all the available task specific data (including the augmented instances). Then, the whole network is trained on a supervised task.

For the evaluation of the context independent method with short texts, we focused on sentiment analysis and the classification of crisis Twitter data. Sentiment analysis will be performed with subsampled SST-2 [50] datasets to simulate a low data regime on a standardized dataset, similar to [15] and [29]. As these constructed conditions are restricted in their real world applicability, we perform further evaluations with real world low data regimes.

The crisis classification tasks have very limited resources, as described by [21]. The first three datasets from [37] are labeled according to whether or not they are informative on the specific topics that are related to the Boston Bombings, the Bohol Earthquake and the West Texas Explosions in 2013. The other two datasets from [46] consist of city-specific Twitter posts that are labeled as incident-related or not.

For the evaluation of the contextual method with long texts, we gathered news articles for topic classification from 2019 and 2020. For the contextual start token we used the respective titles. Topic classification in the news context also faces the problem of few data instances, because the news and dependent topics are often very dynamic and research data is limited. In our case, for every topic, expert groups of two people decided whether the inspected article is relevant to the topic. Furthermore, there is a labeling guideline for every topic so that disagreements are excluded if possible (see appendix B). The topics include three economic issues: layoff, management change (MC) and mergers and acquisitions (M&A) as well as two crisis issues: flood and wildfire.

Before augmenting the data, a fifth of every set was split into a holdout set.

Table 1 Accuracy of the non-contextual text generation process, EDA [59] and the baseline with regard to different SST-2 subsamples (10 runs)

Dataset	Run	Baseline	EDA	Text Gen
SST-2 100	AVG (SD)	0.5581 (0.0463)	0.6934 (0.0124)	**0.7134** (0.0207)
	Best	0.6226	0.7139	**0.7495**
SST-2 300	AVG (SD)	0.7241 (0.0119)	0.7217 (0.0047)	**0.7402** (0.0067)
	Best	0.7417	0.7295	**0.7534**
SST-2 500	AVG (SD)	0.7505 (0.0077)	0.7534 (0.0074)	**0.7598** (0.0126)
	Best	0.7651	0.7671	**0.7754**
SST-2 700	AVG (SD)	**0.7646** (0.0054)	0.7578 (0.0038)	0.7627 (0.0066)
	Best	0.7705	0.7632	**0.7754**

The best values are marked in bold

4.3 Evaluation settings and pre-evaluation of hyperparameters

All data augmentation methods are compared against a baseline with the help of 10-fold training executions. Additionally, the sentiment analysis results are compared with the EDA data augmentation method from [59]. For the sentiment analysis task we augment both classes, while for the others we augment the minority class.

The text generation process offers various possibilities for hyperparameter optimization, which we evaluated with different datasets to avoid overfitting. Ablation studies are shown on the sentiment dataset (Sect. 4.4.1). During generation, the model has a temperature parameter which, the larger it is chosen, the more creative the texts will be and new linguistic patterns occur. However, a too high value can lead to instances that are not topic related. A value that is too low can mean that the model repeats itself very often, while a value too high can result in a loss of the actual theme in the texts [22]. According to the author of the implementation used in this paper, the most suitable value is between 0.7 and 0.9 [60]. In a evaluation with the management change topic, 0.7 was most suitable for the existing case of application, compared to 0.8 and 0.9.

The filtering of generated documents is of importance in this process, as GPT-2 can generate novel instances that may have no relation to the actual class. The parameter of this filtering was chosen individually so that the ten most distant documents would still be marked accordingly to the class. In the ablation studies of Sect. 4.4.1 we show that this filtering is necessary to achieve the high results of this method.

Another aspect that can be added to hyperparameter optimization concerns the number of documents to be generated per training instance. As this is a very important factor we consider it in the evaluations of Sect. 4.4.1.

More details on the hyperparameters of the model can be found in the appendices.

4.4 Results I: sentiment analysis (context independent method)

The results considered good from a human point of view are also reflected in the quantitative evaluation results, that are presented in Table 1. The proposed data augmentation method has almost in every case better results than the baseline and the EDA method by [59]. Particularly it gains the best improvements the less data is available (additive up to 15.53% and 3.56% compared to the baseline and EDA). However, also with the most data the augmentation method reaches in the best run additive performance improvements of 0.49% and 1.22% compared to the baseline and the EDA method.

The method has the highest improvements if less data is available because the prior knowledge of the GPT-2 model is most effective there. The model also produces well written instances which is not the case with the EDA method that sometimes fails to improve the baseline. Furthermore, the rationale of low performing data augmentation methods of [32] comes into play. In contrary to the EDA method, the proposed augmentation algorithm enriches the training data

Fig. 2 Two instances and their transformations by EDA and our method. The first is from the SST-2 task 4.4 and the second from the West Texas Explosion task 4.6. Text passages where each augmentation method attempted to paraphrase the original instance are highlighted in blue; attempted interpolations or introduction of novelties are highlighted in green (color figure online)
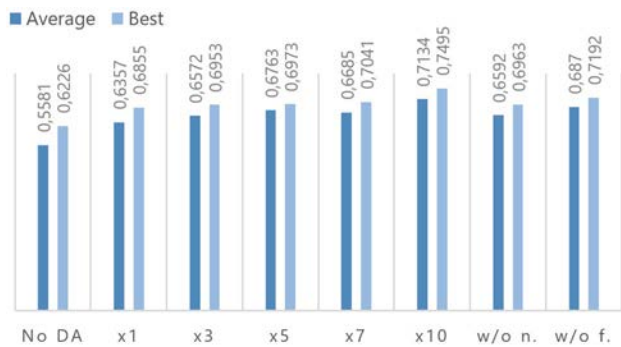
Fig. 3 Evaluation of different augmentation sizes and the omission of the numbering token (w/o n.) and the filtering (w/o f.) step on the SST-2 100 dataset (10 runs)

with new linguistic patterns that have not already been seen by the encoder (Fig. 2).

4.4.1 Ablation studies

Further, we want to show an excerpt of the relevant results from our ablation evaluations (Fig. 3). First, we tested different augmentation sizes that are shown on the SST-2 100 dataset. We limited this evaluation to a maximum of 10 augmentation samples per instance, as the higher numbers demand more computing time. It is evident that the higher the size, the better the results are. A human inspection indicates that even higher numbers might not be as beneficial since the repetition within the samples per instance increases. The human inspection process is detailed in Section F in the Appendix .

Furthermore, we also removed the steps of the augmentation process to see the contributions of each. In a first testing case, we did not include the number of the instance in the finetuning and generation phase (indicated by "w/o n." in Fig. 3). The decrease in the average accuracy by 5.42 points shows that this component is highly important for the whole augmentation process. This also applies to the last step of the augmentation method (indicated by "w/o f." in Fig. 3). Without the manual filtering the average accuracy is reduced by 2.64 points. This insight was already noticed when the filtering parameter was chosen in every task and some instances seemed to be unrelated to the class.

In summary, this indicates that all the steps of the augmentation process need to be included to reach the highest scores. The next evaluation studies are based on this best combination.

Furthermore, we conducted an error analysis by inspecting the generated instances of our technique and comparing them to the EDA method. For our method, it is not clear from which original instance the generation originates, as

it could be an interpolation of more than one instance. Nevertheless, we tried to find the closest original instances by measuring the resemblance by Levenshtein distance.

We selected some insightful examples, which are presented in Fig. 2. The first example shows that EDA is able to keep the label, but substitutes a word that should not be substituted (*oscar wilde play→ academy award wilde play*). On the other hand, our method is able to find suitable words for the replacement and expands the original instance in a meaningful way without distorting the label or the general content. Nevertheless, while we did not see this case in our analysis, it might be possible that the language model is not able to infer the label during finetuning and that the augmentation would change the label. A similar case can be seen regarding the Dublin task of Sect. 4.6, where we assume that the model was not able to infer the label due to a very high diversity in the instances and the label space. While the model does not incorrectly change the label, we see that more than 50% of the generated instances are "???". The more data is generated, the more such instances are created and the content-rich instances are repeated.

The second example displayed in Fig. 2 shows that our method brings a high variation with sensible content but also reformulates "say state officials" to "medical examiner says" which might not be correct. For the task at hand, it is not decisive but it might be with regard to very specific tasks. Moreover, EDA sometimes changes the instance so that it is grammatically incorrect, as it can also be seen in this example. This can be a problem, for example, when using language models that do not expect a specific expression, such as "atomic number", in the context of this instance. Furthermore and even worse, it might happen that the method deletes an essential word, like "not", in the sentence "This movie was not bad", creating an instance with a wrong label when used for sentiment classification.

These examples show that the method proposed in this paper is able to create high-quality and diverse instances. The EDA method instead sometimes creates instances that are wrong or not fitting into the context due to its random character, which is especially critical when using pre-trained language models. Section F in the Appendix provides further examples and analyses of our data augmentation method. It is shown, for example, that the model sometimes completely replicates instances of the training data. This property is not necessarily bad, as it can at least be seen as a sophisticated oversampling method that clones the very important data.

4.5 Results II: news classification (context dependent method)

A qualitative inspection of the data generated for the news dataset shows that high-quality, coherent, and diverse texts have been generated for the different article titles. Nearly

Table 2 Accuracy and F1 scores of the contextual text generation process and the baseline with regard to the five news article topics (10 runs)

Dataset	Run	Accuracy	F1
Layoff	AVG (SD)	0.8350 (0.015)	0.7695 (0.012)
	Best	0.8545	0.7905
With DA	AVG (SD)	**0.8706** (0.009)	**0.8179** (0.011)
	Best	**0.8848**	**0.8354**
MC	AVG (SD)	0.8760 (0.005)	0.7217 (0.021)
	Best	0.8809	0.7627
With DA	AVG (SD)	**0.8853** (0.015)	**0.7559** (0.031)
	Best	**0.9077**	**0.8052**
M&A	AVG (SD)	0.8926 (0.005)	0.6953 (0.011)
	Best	0.8999	0.7075
With DA	AVG (SD)	**0.8975** (0.003)	**0.7095** (0.011)
	Best	**0.8999**	**0.7266**
Flood	AVG (SD)	**0.8462** (0.007)	0.8779 (0.007)
	Best	0.8540	0.8867
With DA	AVG (SD)	0.8408 (0.010)	**0.8804** (0.006)
	Best	**0.8594**	**0.8931**
Wildfire	AVG (SD)	0.9287 (0.016)	0.9253 (0.017)
	Best	**0.9419**	**0.9395**
With DA	AVG (SD)	**0.9312** (0.007)	**0.9297** (0.006)
	Best	0.9395	0.9375

The best values are marked in bold

Table 3 Accuracy and F1 scores of the non-contextual text generation process and the baseline with regard to the five crisis Twitter topics (10 runs)

Dataset	Run	Accuracy	F1
Boston	AVG (SD)	0.7886 (0.019)	0.7344 (0.030)
Bombings	Best	0.8062	0.7720
With DA	AVG (SD)	**0.8003** (0.021)	**0.7588** (0.024)
	Best	**0.8311**	**0.7979**
Bohol	AVG (SD)	0.9097 (0.014)	0.8857 (0.021)
Earthquake	Best	0.9302	0.9126
With DA	AVG (SD)	**0.9238** (0.011)	**0.9062** (0.015)
	Best	**0.9399**	**0.9277**
West Texas	AVG (SD)	0.8486 (0.020)	0.8340 (0.025)
Explosion	Best	0.8804	0.8765
With DA	AVG (SD)	**0.8755** (0.010)	**0.8721** (0.011)
	Best	**0.9004**	**0.8970**
Dublin	AVG (SD)	**0.9893** (0.002)	**0.9199** (0.015)
	Best	**0.9912**	**0.9351**
With DA	AVG (SD)	0.9858 (0.002)	0.8945 (0.014)
	Best	0.9878	0.9116
New York	AVG (SD)	0.9302 (0.016)	0.8428 (0.027)
City	Best	**0.9463**	**0.8701**
With DA	AVG (SD)	**0.9346** (0.003)	**0.8472** (0.007)
	Best	0.9385	0.8555

The best values are marked in bold

all instances had a clear reference to the actual class. In the filtering step, primarily instances where the GPT-2 model frequently repeated words were sorted out.

The classification results of our contextual approach are presented in Table 2 in comparison to the baseline. Looking at the overall evaluation across all topics, it can be seen that the classifiers also achieve very good results through the contextual data augmentation method. Especially for the economic tasks, relative increases in the maximum F1 value of over 4% for the MC and layoff topic and 2% for the M&A task can be recorded. The M&A evaluation illustrates the importance of the F1 measure. While good improvements in accuracy were also achieved for management change and layoff, there is virtually no change in the M&A topic. However, the more significant F1 measure increases substantially. A small improvement is also evident for the flood topic (+0.64 with the maximum F1 value). For the wildfire topic, the text generation approach does not achieve better results. The values are almost the same as for the baseline. However, the standard deviations of the results from the baseline are more than twice as high as with the data augmentation method.

It can be ascertained that the contextual text generation approach is very well suited for the present topic classification tasks. A possible bias regarding the economic topics

could be attributed to the pre-trained GPT-2 model. The model was trained with documents from outgoing Reddit links. There may be comparatively few crisis data among the approximately 8 million documents so that the model is less able to represent this topic area. Another possible explanation for the smaller values of the crisis topics is the already very high classification quality of the two tasks. The flood topic, which was chosen because of the poorer classification quality compared to the wildfire topic, still exceeds the three economic topics in the F1-measure. The wildfire topic is by far the best task for the classifiers (about +5% F1-measure compared to the flood task). An improvement of the two topics may not be possible anymore because the dataset can contain errors as some labelling decisions are difficult and subjective.

4.6 Results III: crisis informatics (context independent method)

In the last section, we stated that GPT-2 might be less usable for crisis data. Since the usage of machine learning in crisis situations is very promising and getting good models is an ongoing issue due to little data and the challenge of domain adaption, we further examine this consideration by focusing only on crisis data for the second evaluation.

Inspecting the newly generated data, we see that the model often produces identical outputs for different runs. The generated data of the context dependent models of the second evaluation is clearly more diverse. However, the context independent method also seems to have the potential to perform very well, as Table 3 shows. It is especially beneficial for the classification tasks of Olteanu et al. (2015) (first three datasets). The averages and best runs outperform the baseline additively by 2.1% to 3.8% and 1.5% to 2.5% in the F1-measure. For the tasks of [46], however, the method does not provide any substantial improvements regarding the direct scores. For the Dublin dataset, the augmentation method even seems to have a negative effect. Yet, a decrease in the F1 standard deviation is achieved for every task.

Although we can confirm the assumption of the sentiment analysis that the context independent variant of the augmentation method creates less diverse instances, it analogously has a very positive impact on the classification quality when applied to real world low data regimes. Especially the narrowly defined problems by [37] are well suited, leading to the consideration that the difference of the results lies in the nature of the problems. While the first three tasks are bond to a special crisis event, the two other tasks are just incident related with no other focus than the respective city. It may be that these two tasks are too broadly defined with very different instances so that the model was not able to properly finetune to generate sensible instances. Apart from that, the gain in robustness is clearly visible because a decrease in the F1 standard deviation is achieved on each task. The evaluation also shows that our data augmentation methods can achieve good results not only on economic topics.

5 Discussion and conclusion

While there are numerous beneficial data augmentation methods in computer vision, textual transformations are more difficult to define [24, 59] and often result in mixed results [32]. In order to address these issues, we presented two data augmentation methods for long and short texts based on text generation techniques to enhance the knowledge base on small data analytics. Our results on the 11 datasets, which are listed in an aggregated form in Table 4, contribute to answering the following research questions.

How can we utilize text generation approaches of data augmentation that achieve a high novelty in the data while preserving the label quality to improve pre-trained machine learning classifiers (RQ1)?

We proposed two data augmentation methods that are based on text generating language models. We constructed three different steps (see Fig. 1) for ensuring a high label preservation within the transformations of these models. As a first step, the data was primed by a special token.

Table 4 Average and maximum F1 performance deltas of the data augmentation models in comparison to their respective baseline counterparts across all datasets

Dataset	Delta Avg.	Delta Max
SST-2 (100)-Acc.	+15.53%	+12.69%
SST-2 (300)-Acc.	+1.61%	+1.17%
SST-2 (500)-Acc.	+0.93%	+1.03%
SST-2 (700)-Acc.	−0.19%	+0.49%
Layoff-F1	+4.84%	+4.49%
MC-F1	+3.42%	+4.25%
M&A-F1	+1.42%	+1.91%
Flood-F1	+0.25%	+0.64%
Wildfire-F1	+0.44%	−0.20%
Boston Bombings-F1	+2.44%	+2.59%
Bohol Earthquake-F1	+2.05%	+1.51%
West Texas Explosion-F1	+3.81%	+2.05%
Dublin-F1	−2.54%	−2.35%
New York City-F1	+0.44%	−1.46%

This token can signal the model to generate training data for this class in the generation phase. In order to ensure that the model is familiar with the class data and the token, finetuning was carried out with the prepared data in the second step. After the generation, filtering based on the BERT document embeddings [41] formed the final step. In our experimental setup, we used a pre-trained encoder and finetuned it on the various tasks, including the augmented data, to face the challenge of creating a sophisticated data augmentation method that also performs well on pre-trained models. The two derived methods achieved very good results in the evaluation phase with several performance gains and reductions of the standard deviations.

When evaluating the utility of the algorithm, however, further criteria must be considered. While it is rather easy to embed the augmentation method into a classification process, the GPT-2 model needs some time to be executed. The generation of one example of a long dataset took about 10–30 s. Noise-inducing methods such as EDA take much less than a second to complete an instance, as they only perform simple operations such as changing the order of words, deleting some words, or generating misspellings [6, 10, 16, 38]. However, text generation methods such as ours are limited by the time required by the generation process. Nevertheless, compared to the time it takes a human to label a new instance, our method is still very advantageous. The time required can also be significantly reduced, for example, by using a different language model that is faster. Furthermore, the used GPT-2 model is mainly limited to English, making it less usable for multilingual tasks. However, this can also be mitigated by using

another language model as the proposed method is suitable for different language models.

In which way is the incorporation of contexts of long text instances in classification problems helpful when using text generation as data augmentation method (RQ1.1)?

When dealing with long texts, we decided to integrate a context-based token in the generation phase so that the generated texts are more explicit for the respective instance and highly diverse among all instances. A closer look into the generated samples from the evaluation phase confirmed this assumption. Furthermore, the generated instances seem to be very coherent and related to the task at hand, due to the strengths of the GPT-2 language model. More importantly, four of the five tasks could be improved by including the newly generated and filtered instances. This led to an additive increase in the average and maximum F1 value of up to 4.8% and 4.5% respectively. However, we noticed that the augmentation technique could not improve the classification results when the classifier already performs very well without additional data.

How is it possible to achieve a quality improvement for classification tasks with short texts when augmenting with text generation (RQ1.2)?

For classification tasks with short texts, a context-based token integration is not possible, wherefore we included the number at which the respective instance occurred in the fine-tuning. On closer inspection of the newly created instances, several duplicates were found. This did not have negative implications for the evaluation, since repeating some training examples resembles the process of simple oversampling. It may even be interpreted as a more sophisticated version, where some examples are completely new and the others are oversampled from the most fitting data points. Accordingly, a great performance gain could be achieved in the constructed and real world low data regimes, leading to improvements of up to 15.53 and 3.81 points respectively. We noticed that this augmentation method was not suitable for two special real world tasks. We hypothesize that these two tasks are too broadly defined on the rationale that the GPT-2 model is not able to infer the right context just based on the finetuning of the data.

5.1 Empirical, practical and theoretical contributions

Considering our findings, the study revealed practical, theoretical and empirical contributions:

New data augmentation methods based on the GPT-2 language model. The evaluation results of the data augmentation methods indicated that the GPT-2-model, in combination with three safety steps, can achieve a considerable improvement in the text classification tasks (see Tables 1 to 3). In contrast to the similar approaches of [57] and [3] that utilized GPT-2 for text generation too, our method is more generally applicable and offers more safety steps. Ref. [57] describe no measures for label preservation in their approach, and [3] only enable data augmentation for instances consisting of one sentence. Furthermore, we include a filtering mechanism which includes the human expertise, strongly increasing the diversity without much supervision. The advantages of the text generation approach proposed here facilitate a wide use, qualifying it as basic element for further adaption in prospective classification applications.

A textual data augmentation basis which is beneficial for pre-trained classification models. Ref. [32] show that data augmentation might not be helpful when dealing with state-of-the-art pre-trained models. This seems logical from a theoretical perspective since pre-training and the transfer to new tasks also follow the goal of reducing the amount of necessary training data. In order to get an enrichment anyway, sophisticated augmentation methods are needed, which should provide unseen linguistic patterns that are relevant to the task [32]. We addressed this issue by leveraging the GPT-2 model that was trained with more than 8 million web pages. This gives the great potential to include new linguistic patterns in the generated data (example instances can be found in Appendix F). For creating task relevant data, we derived three steps that increase the possibility of class related content. In the evaluation we showed that the proposed method is able to improve the pre-trained encoder model. In contrast to [32], we did not test the method on a transformer model. However, we trained the pre-trained ULMFit encoder for both testing cases (no augmentation and augmentation) with the augmented data so that for the encoder no data is unseen beforehand.

Empirical insights into the domain-specific application of small data analytics. In this work, we gathered new empirical insights into the application of data augmentation in the research domains of sentiment analysis, news classification, and crisis informatics. In crisis informatics, various studies have examined the use of domain adaptation, transfer learning, active learning, and online learning to reduce the labeling effort [17, 21, 36]. However, few research has examined the application of textual data augmentation for crisis management [57], which we enhance by the evaluation and interpretation of seven augmented datasets. For sentiment analysis we constructed a low data regime, like [28] and [15]. Small data analysis research is gaining popularity and there is a need to establish datasets that can be used, understood and compared by all types of researchers. We strengthen this research direction by basing our evaluation on this dataset. Furthermore,

with the five news classification datasets we are exploring the important topic of long text classification, that is often neglected in research. News classification is mostly done with short descriptions of the articles only, as in the AG News dataset [66].

5.2 Limitations and outlook

During the evaluation, a restriction was made with regard to the language model used. While applicable with other language models, it is not clear if the performance gain remains the same. The GPT-3 model by [7] seems to be the next sensible choice for increasing the results. However, the finetuning step that is likely to be necessary is currently not possible due to the high resource utilization of the model. Nonetheless, because of its size and linguistic expressiveness, it may be especially helpful to address the challenge stated by [32] in which pre-trained models might not gain any improvement from data augmentation. In relation to the study by [32], future research could test the proposed method with transformer models. It would also be interesting to see how smaller language models perform, that may be much faster. In addition, there might also be an option to fully automate the filtering step, which further increases the universal usability, even if the human effort is already very low now.

Despite many efforts in data augmentation, the big data wall problem addressed at the beginning is still of great relevance. However, if in the future, according to various assumptions, very large models, such as GPT-3 by [7], are better able to solve these problems, the high resource wall problem opens up, which only allows large companies to train and use these models.

Appendices

A. Availability of data and material

In summary, most datasets analyzed during the current study are publicly available. In the **first evaluation** (Sect. 4.4), we used the SST dataset[1] of [50] which is publicly available.

The datasets analyzed during the **second evaluation** (Sect. 4.5) are not publicly available due to publication restrictions by news agencies. Still, a concise description of these datasets is given in Appendix B. We would like to highlight that news classification is a highly relevant field in the industry that receives too little attention in academia. In addition, we had to decide to create our own dataset in order to tackle long text classification, as these datasets are particularly rare. We hope that we have included enough results from the public datasets so that reproducibility can be inferred.

For the **third evaluation** (Sect. 4.6), three public datasets (Boston Bombings, the Bohol Earthquake and the West Texas Explosions) from the CrisisLexT26[2] annotated data groups from [37] and two public datasets (Dublin and New York City) from the annotated data groups from [46][3] were used. The primary language of all datasets is English.

B. Description of the datasets used during the second evaluation

The datasets consist of English news articles of over 2600 different source domains from the years 2019 and 2020, which were preselected with regard to specific query words. For each topic, the articles received values on the basis of these query words to split them into 12 different buckets. We sampled the news articles from these buckets uniformly so that highly diverse instances were labeled. A short summary of the labeling guidelines and data distributions are described in the following list:

Layoff. The layoff topic consists of all forms of dismissals of employees in the corporate context. A total of 1992 articles were annotated of which 751 instances are positive and 1241 are negative.

Management change. This topic covers all forms of changes (retirement, resignation, appointment) of the board of directors and important positions in companies, organizations and advisory boards. 2129 instances were labeled of which 567 instances are positive and 1562 are negative.

Mergers and Acquisitions. M&A includes all transactions in which ownership is transferred to companies or their operating units. The mere investment in a company is seen as negative here. For this topic, 2227 instances were labeled of which 474 are positive and 1753 are negative.

Flood. The flooding topic is positively recognized in news if the article deals with the actual flooding or the main topic is a consequence of a flood. If only an increase in the water level is reported, this message should not be regarded as positive. There are 2533 identified instances on this topic, 1639 of which are positive and 894 are negative. Although more positive than negative instances were identified here, the negative class is still seen as the majority class since the clear majority of all messages on the Internet are not related to floods.

Wildfire. All forms of wildfire are classified under this topic. House fires and metaphorical uses of the term are labeled negatively. The dataset includes 2410 identified

---

[1] SST datasets of [50]: https://nlp.stanford.edu/sentiment/index.html

[2] CrisisLexT26 datasets from [37]: https://github.com/sajao/CrisisLex/tree/master/data/CrisisLexT26

[3] Datasets from [46]: http://www.doc.gold.ac.uk/~cguck001/IncidentTweets/

instances of which 1202 are positive and 1208 are negative. Similar to the flood topic, the negative class is the majority class.

In the preprocessing step we added the tokens "xxtitle" before every title and "xxbodytext" before the start of the normal article text.

C. Algorithm

In the following, we defined the algorithm of the context independent data augmentation method.

Algorithm 1: Augmentation of Short Texts

Input: Language Model LM, Class data X_c,
   Document Embedding Model E,
   Number of instances per training data n

1. For each instance in X_c: Attach
   '$< |startoftext| > |i|$' to the beginning and
   '$< |endoftext| >$' to the end of the ith instance to
   obtain X_{cprep}
2. Finetune LM on X_{cprep} to obtain LM_{cprep}
3. For each k in $|X_{cprep}|$: Generate n new instances
   with LM_{cprep} and '$< |startoftext| > |i|$' as prefix
   to obtain X_{gen}
4. Embed all instances in X_{gen} and X_c with E
5. Obtain $X_{filtered}$ by including all instances of
   X_{gen} at which the embedding representations are
   close to the centroid of the embedded X_c

Result: $X_{filtered}$

Algorithm 2: Augmentation of Long Texts

Input: Language Model LM, Class data X_c,
   Document Embedding Model E,
   Number of instances per training data n,
   Function extracting context part $cont()$

1. For each instance in X_c: Attach
   '$< |startoftext| >$' to the beginning and
   '$< |endoftext| >$' to the end of the instances to
   obtain X_{cprep}
2. Finetune LM on X_{cprep} to obtain LM_{cprep}
3. For each t in X_{cprep}: Generate n new instances
   with LM_{cprep} and '$< |startoftext| >$' $+ cont(t)$ as
   prefix to obtain X_{gen}
4. Embed all instances in X_{gen} and X_c with E
5. Obtain $X_{filtered}$ by including all instances of
   X_{gen} at which the embedding representations are
   close to the centroid of the embedded X_c

Result: $X_{filtered}$

D. Ethics

In our work, we have placed particular emphasis on ethics and repeatedly reassessed our approach with regard to responsibilities. We have restricted ourselves to only using the textual content and the labels in the datasets to respect privacy as much as possible. When working with social media data, we especially did not use, aggregate or draw any conclusion from further metadata such as the name or location of a user.

For the practical implementation of our method, we want to mention that the GPT-2 model as well as most other language models contain biases (for example a gender, religious or racial bias) [52]. Using the method in a real application can result in a domain shift and/or an inclusion of those biases. This can explicitly lead to discriminatory decisions by the machine learning model, even if the dataset itself does not contain any bias.

E. Architecture, hyperparameters and infrastructure

In the evaluation, we used the pre-trained ULMFit model by [13] as implemented in fastai [12]. The model consists of a pre-trained encoder that is based on the AWD-LSTM architecture by [34] and a linear pooling classifier. The classifier consists of a layer that concatenates the final outputs of the encoder with the maximum and the average of all intermediate outputs and two fully connected layers.[4] Further information about the general architecture can be extracted from the paper of [13] and the implementation in fastai [12]. The encoder finetuning is done by preparing all available data of the respective task (including the augmented data) for the language modeling task. In the training, we performed 15 cycles with the 1cycle policy by [49]. We used a learning rate of 0.002 and a maximum and minimum momentum of 0.8 and 0.7. Overall, we used a fixed batch size of 64 and a backpropagation through time window of 70. Each encoder and classifier was trained to the downstream task with three cycles with gradual unfreezing and another five cycles with the unfrozen model. The learning rate was individually determined by the learning rate range test by [49] with a range from 10^{-7} to 10 over 100 iterations.[5] On the outputs we used label smoothing with an epsilon parameter of 0.1. As optimization algorithm, we used the Adam algorithm [23].

For the text generation, we used the gpt-2-simple implementation[6] to finetune and generate texts from the GPT-2 model. Only the parameters discussed in chapter 4.2 were varied. For the filtering step, Sentence-BERT[7] [41] with the "roberta-large-nli-stsb-mean-tokens" transformer model was used.

---

[4] ULMFit implementation: https://fastai1.fast.ai/text.learner.html#text_classifier_learner

[5] Learning rate range test implementation: https://fastai1.fast.ai/callbacks.lr_finder.html#callbacks.lr_finder

[6] gpt-2-simple implementation: https://github.com/minimaxir/gpt-2-simple

[7] Sentence-BERT implementation: https://github.com/UKPLab/sentence-transformers

Table 5 Generated data instances and their most similar original counterparts

West Texas Explosions	Media Matters Texas: West, Texas, fertilizer plant: 28 years, no full inspection: Before 270 tons of ammonium nitrate exploded at ... http://t.co/XwUwgRcGDT
	West, Texas, fertilizer plant: 28 years, no full inspection: Before 270 tons of ammonium nitratet exploded at ... http://t.co/XwUwgRcGDT
	RT @mpoindc: I want them to know that they are not forgotten, Obama says to victims of explosion in West, TX.
	I want them to know that they are not forgotten, Obama says to victims of explosion in West, TX.
	RT @BreakingNews: Large explosion reported at fertilizer plant near Waco, Texas—@CBSDFW http://t.co/xOIyCyuxFD
	Explosion reported at fertilizer plant near Waco, Texas—@CBSDFW http://t.co/xOIyCyuxFD
	RT @BBCBreaking: 12 confirmed dead, approximately 200 injured in #West fertiliser plant explosion in Texas, say state officials
	Searchers Find 12 Bodies After Texas Explosion http://t.co/fnBc2LmXs (CNN)—Hundreds believed injured in Texas fertilizer plant explosion, medical examiner says. http://t .
	RT @laurenonizzle: This iPhone video of the fertilizer plant explosion near #Waco will send chills up your spine. Surreal. http://t.co/O .
	RT @laurenonizzle: This iPhone video of the fertilizer plant explosion near #Waco will send chills up your spine. Surreal. http://t.co/O .
SST-2 (100)	Smart, sassy interpretation of the oscar wilde play.
	Harp, sassy interpretation of the oscar wilde play , with an unexpected twist.
	A fast paced and suspenseful argentinian thriller about the shadow side of play.
	A fast paced and suspenseful argentinian thriller.
	This comic gem is as delightful as it is derivative.
	The film is bright and flashy in all the right ways.
	The best movie in many a moon about the passions that sometimes fuel our best achievements and other times leave us stranded with nothing more than our lesser appetites.
	The best film in many a moon about the passions that sometimes fuel our best achievements and other times leave us stranded with nothing more than our lesser appetites.
	A fine production with splendid singing by angela gheorghiu , ruggero raimondi , and roberto alagna.
	A fine production with splendid singing by angela gheorghiu , ruggero raimondi , and roberto alagna.

We tokenized all datasets with the built-in tokenizers of the gpt-2-simple, Sentence-BERT and fastai libraries for the different use cases.

The evaluation of chapter 4.4 was performed on a Nvidia Quadro RTX 6000 graphics card with 24 GB RAM. For the evaluation of chapter 4.3 and 4.5, less resources were necessary, which is why a Nvidia Tesla P100 with 16GB RAM was used.

In general, the tasks were quite resource intensive. The finetuning of the GPT-2 model on the flood dataset took about five hours, while the generation of ten examples per instance occupied about four days. The finetuning of the language model and the training of the classifier together took about another four hours.

F. Generated data

see Table 5

In our experiments, we also analyzed the generated data. For this purpose, we first selected generated instances and tried to find the original instance with the closest resemblance (measured by Levenshtein distance). In Table 5 a excerpt of some instances, original and generated counterparts, from the West Texas Explosions and SST-2 datasets are given. Here we can see, that the GPT-2 model is, for example, able to remove preceding words (first and second instance of West Texas Explosions) or even truncate the sentence at the end (second instance of SST-2). The first example of the SST-2 dataset shows that the model is also able to enlarge the original instance. In case of the third example of West Texas Explosions it could be that the original instance is interpolated with another instance. The fourth instance of the SST-2 dataset also shows that the model can make small changes like synonym substitution. Furthermore, for many generated instances we were not able to find similar counterparts, see for example the third instance of the SST-2 dataset given in Table 5. These can, for example, be instances the model learned beforehand, which supports the consideration that the model is able to create highly diverse examples with new linguistic features. However, as shown with the last examples of the two datasets, the model also sometimes repeats the original instance. We also noticed that the more data we generate, the more duplicates can be found. This is reasonable as the probability for some token sequences is very high and they are therefore generated more often. As already indicated to some extent in the ablation studies, it would be interesting to see how many artificial instances can be generated till a maximum improvement is reached and whether this is dependent on the size of the language model. In general, the findings indicate that the method proposed in our study is capable of performing many different transformations.

Availability of data and material The availability of data is outlined in Appendix A.

Declarations

Conflicts of interest/Competing interests Not applicable.

Code availability The code specifics are outlined in Appendix C.

Ethics Ethics are discussed in Appendix D.

References

1. Alam F, Ofli F, Imran M (2020) Descriptive and visual summaries of disaster events using artificial intelligence techniques: case studies of hurricanes harvey, irma, and maria. Behav Inf Technol 39(3):288–318. https://doi.org/10.1080/0144929X.2019.1610908

2. Alzantot M, Sharma Y, Elgohary A, Ho BJ, Srivastava MB, Chang KW (2018) Generating natural language adversarial examples. In: Proceedings of EMNLP. https://doi.org/10.18653/v1/d18-1316

3. Anaby-Tavor A, Carmeli B, Goldbraich E, Kantor A, Kour G, Shlomov S, Tepper N, Zwerdling N (2020) Do not have enough data? Deep learning to the rescue! Proceedings of the AAAI. http://arxiv.org/abs/1911.03118

4. Banko M, Brill E (2001) Scaling to very very large corpora for natural language disambiguation. In: Proceedings of the 39th annual meeting of the Association for Computational Linguistics. https://doi.org/10.3115/1073012.1073017

5. Bayer M, Kaufhold MA, Reuter C (2021) A survey on data augmentation for text classification. https://arxiv.org/abs/2107.03158

6. Belinkov Y, Bisk Y (2018) Synthetic and natural noise both break neural machine translation. In: Proceedings of ICLR

7. Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler DM, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D (2020) Language models are few-shot learners. In: NeurIPS, http://arxiv.org/abs/2005.14165

8. Carreira R, Crato JM, Gonçalves D, Jorge JA (2004) Evaluating adaptive user profiles for news classification. In: Proceedings IUI. https://doi.org/10.1145/964442.964481

9. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. JAIR.https://doi.org/10.1613/jair.953

10. Coulombe C (2018) Text data augmentation made simple by leveraging NLP cloud APIs. arXiv preprint arXiv:1812.04718, pp 1–33. http://arxiv.org/abs/1812.04718

11. Fadaee M, Bisazza A, Monz C (2017) Data augmentation for low-resource neural machine translation. In: ACL. https://doi.org/10.18653/v1/P17-2090

12. Howard J, Gugger S (2020) Fastai: a layered api for deep learning. Information (Switzerland). https://doi.org/10.3390/info11020108

13. Howard J, Ruder S (2018) Universal language model fine-tuning for text classification. In: Proceedings of ACL. https://doi.org/10.18653/v1/p18-1031

14. Hu YQ, Yu Y (2020) A technical view on neural architecture search. Int J Mach Learn Cybern 11(4):795–811. https://doi.org/10.1007/s13042-020-01062-1

15. Hu Z, Tan B, Salakhutdinov R, Mitchell T, Xing EP (2019) Learning data manipulation for augmentation and weighting

16. Huong TH, Hoang VT (2020) A data augmentation technique based on text for Vietnamese sentiment analysis. In: Proceedings of IAIT pp 1–5. https://doi.org/10.1145/3406601.3406618

17. Imran M, Castillo C, Diaz F, Vieweg S (2018) Processing social media messages in mass emergency: Survey summary. In: Companion proceedings of the the web conference 2018, international world wide web conferences steering committee, Republic and Canton of Geneva, CHE, WWW '18, pp 507–511. https://doi.org/10.1145/3184558.3186242

18. Jiao X, Yin Y, Shang L, Jiang X, Chen X, Li L, Wang F, Liu Q (2019) TinyBERT: distilling BERT for natural language understanding. In: EMNLP 2020, pp 1–14. http://arxiv.org/abs/1909.10351

19. Kafle K, Yousefhussien M, Kanan C (2018) Data augmentation for visual question answering. In: Proceedings of the 10th international conference on natural language generation. https://doi.org/10.18653/v1/w17-3529

20. Kaufhold MA (2021) Information refinement technologies for crisis informatics: user expectations and design principles for social media and mobile apps. Springer Verlag, Wiesbaden, Germany

21. Kaufhold MA, Bayer M, Reuter C (2020) Rapid relevance classification of social media posts in disasters and emergencies: a system and evaluation featuring active, incremental and online learning. Inf Process Manage. https://doi.org/10.1016/j.ipm.2019.102132

22. Khan B (2019) Generate your own text with OpenAI's GPT-2. https://www.kaggle.com/bkkaggle/generate-your-own-text-with-openai-s-gpt-2-117m

23. Kingma DP, Ba JL (2015) Adam: a method for stochastic optimization. In: ICLR 2015—conference track proceedings

24. Kobayashi S (2018) Contextual augmentation: data augmentation by words with paradigmatic relations. arXiv preprint arXiv:1805.06201. https://doi.org/10.18653/v1/n18-2072

25. Kolomiyets O, Bethard S, Moens MF (2011) Model-portability experiments for textual temporal analysis. In: Proceedings of ACL-HLT

26. Krishnalal G, Rengarajan SB, Srinivasagan KG (2010) A new text mining approach based on HMM-SVM for web news classification. Int J Comput Appl. https://doi.org/10.5120/395-589

27. Kruspe A, Kersten J, Wiegmann M, Stein B, Klan F (2018) Classification of incident-related tweets : tackling imbalanced training data using hybrid CNNs and translation-based data augmentation. In: Notebook papers of TREC

28. Kumar A, Bhattamishra S, Bhandari M, Talukdar P (2019) Submodular optimization-based diverse paraphrasing and its effectiveness in data augmentation. In: Proceedings of NAACL-HLT, pp 3609–3619. https://doi.org/10.18653/v1/n19-1363

29. Kumar V, Choudhary A, Cho E (2020) Data augmentation using pre-trained transformer models

30. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 10(1109/5):726791

31. Liu B, Zhang L (2012) A survey of opinion mining and sentiment analysis. Springer, Boston, MA, US, pp 415–463

32. Longpre S, Wang Y, DuBois C (2020) How effective is task-agnostic data augmentation for pretrained transformers? In: Findings of EMNLP

33. Medhat W, Hassan A, Korashy H (2014) Sentiment analysis algorithms and applications: a survey. Ain Shams Eng J 5(4):1093–1113. https://doi.org/10.1016/j.asej.2014.04.011

34. Merity S, Keskar NS, Socher R (2018) Regularizing and optimizing LSTM language models. In: ICLR 2018—conference track proceedings

35. Miyato T, Dai AM, Goodfellow I (2017) Adversarial training methods for semi-supervised text classification. In: Conference Track - ICLR

36. Nguyen D, Ali Al Mannai K, Joty S, Sajjad H, Imran M, Mitra P (2017) Robust classification of crisis-related data on social networks using convolutional neural networks. In: Proceedings of the international AAAI conference on web and social media 11(1). https://ojs.aaai.org/index.php/ICWSM/article/view/14950

37. Olteanu A, Vieweg S, Castillo C (2015) What to expect when the unexpected happens: social media communications across crises. In: Proceedings of CSCW. https://doi.org/10.1145/2675133.2675242

38. Qiu S, Xu B, Zhang J, Wang Y, Shen X, de Melo G, Long C, Li X (2020) EasyAug: an automatic textual data augmentation platform for classification tasks. In: Companion proceedings of the web conference 2020. https://doi.org/10.1145/3366424.3383552

39. Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I (2018) Language models are unsupervised multitask learners. In: OpenAI blog

40. Raghuwanshi BS, Shukla S (2021) Classifying imbalanced data using SMOTE based class-specific kernelized ELM. Int J Mach Learn Cybern 12(5):1255–1280. https://doi.org/10.1007/s13042-020-01232-1

41. Reimers N, Gurevych I (2019) Sentence-BERT: sentence embeddings using Siamese BERT-networks. 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). https://doi.org/10.18653/v1/d19-1410

42. Reuter C, Marx A, Pipek V (2012) Crisis management 2.0: towards a systematization of social software use in crisis situations. Int J Inf Syst Crisis Response Manage (IJISCRAM) 4(1):1–16. https://doi.org/10.4018/jiscrm.2012010101

43. Reuter C, Ludwig T, Kaufhold MA, Spielhofer T (2016) Emergency services attitudes towards social media: a quantitative and qualitative survey across europe. Int J Hum Comput Stud (IJHCS) 95:96–111. https://doi.org/10.1016/j.ijhcs.2016.03.005

44. Rizos G, Hemker K, Schuller B (2019) Augment to prevent: short-text data augmentation in deep learning for hate-speech

classification. In: Proceedings of CIKM. https://doi.org/10.1145/3357384.3358040

45. Şahin GG, Steedman M (2018) Data augmentation via dependency tree morphing for low-resource languages. In: Proceedings of the 2018 conference on empirical methods in natural language processing. https://doi.org/10.18653/v1/d18-1545

46. Schulz A, Guckelsberger C, Janssen F (2017) Semantic abstraction for generalization of tweet classification: an evaluation of incident-related tweets. Semantic Web. https://doi.org/10.3233/SW-150188

47. Sennrich R, Haddow B, Birch A (2016) Improving neural machine translation models with monolingual data. In: ACL, https://doi.org/10.18653/v1/p16-1009

48. Shorten C, Khoshgoftaar TM (2019) A survey on image data augmentation for deep learning. J Big Data. https://doi.org/10.1186/s40537-019-0197-0

49. Smith LN (2018) A disciplined approach to neural network hyperparameters: Part 1- learning rate, batch size, momentum, and weight decay

50. Socher R, Perelygin A, Wu JY, Chuang J, Manning CD, Ng AY, Potts C (2013) Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of EMNLP

51. Soden R, Palen L (2018) Informating crisis: Expanding critical perspectives in crisis informatics. In: Proc ACM Hum-Comput Interact 2 (CSCW). https://doi.org/10.1145/3274431

52. Solaiman I, Brundage M, Clark J, Askell A, Herbert-Voss A, Wu J, Radford A, Wang J (2019) Release strategies and the social impacts of language models

53. Stieglitz S, Mirbabaie M, Ross B, Neuberger C (2018) Social media analytics—challenges in topic discovery, data collection, and data preparation. Int J Inf Manage 39:156–168

54. Sun X, He J (2020) A novel approach to generate a large scale of supervised data for short text sentiment analysis. multimedia tools and applications. https://doi.org/10.1007/s11042-018-5748-4

55. Sun C, Shrivastava A, Singh S, Gupta A (2017) Revisiting unreasonable effectiveness of data in deep learning era. In: Proceedings of the ICCV. https://doi.org/10.1109/ICCV.2017.97

56. Taylor L, Nitschke G (2019) Improving deep learning with generic data augmentation. In: Proceedings of SSCI. https://doi.org/10.1109/SSCI.2018.8628742

57. Wang C, Lillis D (2020) Classification for crisis-related tweets leveraging word embeddings and data augmentation. In: TREC 2019. https://trec.nist.gov/

58. Wang WY, Yang D (2015) That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using #petpeeve tweets. In: Proceedings of EMNLP. https://doi.org/10.18653/v1/d15-1306

59. Wei J, Zou K (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). https://doi.org/10.18653/v1/d19-1670

60. Woolf M (2019) GitHub—gpt-2-simple: Python package to easily retrain OpenAI's GPT-2 text-generating model on new texts. https://github.com/minimaxir/gpt-2-simple

61. Xiang R, Chersoni E, Lu Q, Huang CR, Li W, Long Y (2021) Lexical data augmentation for sentiment analysis. J Assoc Inf Sci Technol 72(11):1432–1447. https://doi.org/10.1002/asi.24493

62. Xu Y, Jia R, Mou L, Li G, Chen Y, Lu Y, Jin Z (2016) Improved relation classification by deep recurrent neural networks with data augmentation. In: Proceedings of COLING 2016: technical papers

63. Zeiler MD, Fergus R (2013) Stochastic pooling for regularization of deep convolutional neural networks. In: Proceedings of ICLR

64. Zhai J, Qi J, Zhang S (2021) Imbalanced data classification based on diverse sample generation and classifier fusion. Int J Mach Learn Cybern. https://doi.org/10.1007/s13042-021-01321-9

65. Zhang H, Cisse M, Dauphin YN, Lopez-Paz D (2018) MixUp: beyond empirical risk minimization. In: Conference track of ICLR

66. Zhang X, Zhao J, Lecun Y (2015) Character-level convolutional networks for text classification. In: NIPS