future internet

MDPI

Article

Use of Data Augmentation Techniques in Detection of Antisocial Behavior Using Deep Learning Methods

Viera Maslej-Krešňáková, Martin Sarnovský * and Júlia Jacková

Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Technical University of Košice, 040 01 Kosice, Slovakia
* Correspondence: martin.sarnovsky@tuke.sk

Abstract: The work presented in this paper focuses on the use of data augmentation techniques applied in the domain of the detection of antisocial behavior. Data augmentation is a frequently used approach to overcome issues related to the lack of data or problems related to imbalanced classes. Such techniques are used to generate artificial data samples used to improve the volume of the training set or to balance the target distribution. In the antisocial behavior detection domain, we frequently face both issues, the lack of quality labeled data as well as class imbalance. As the majority of the data in this domain is textual, we must consider augmentation methods suitable for NLP tasks. Easy data augmentation (EDA) represents a group of such methods utilizing simple text transformations to create the new, artificial samples. Our main motivation is to explore EDA techniques' usability on the selected tasks from the antisocial behavior detection domain. We focus on the class imbalance problem and apply EDA techniques to two problems: fake news and toxic comments classification. In both cases, we train the convolutional neural networks classifier and compare its performance on the original and EDA-extended datasets. EDA techniques prove to be very task-dependent, with certain limitations resulting from the data they are applied on. The model's performance on the extended toxic comments dataset did improve only marginally, gaining only 0.01 improvement in the F1 metric when applying only a subset of EDA methods. EDA techniques in this case were not suitable enough to handle texts written in more informal language. On the other hand, on the fake news dataset, the performance was improved more significantly, boosting the F1 score by 0.1. Improvement was most significant in the prediction of the minor class, where F1 improved from 0.67 to 0.86.

Keywords: data augmentation; EDA; deep learning; antisocial behavior; fake news detection; toxic comments

Citation: Maslej-Krešňáková, V.; Sarnovský, M.; Jacková, J. Use of Data Augmentation Techniques in Detection of Antisocial Behavior Using Deep Learning Methods. *Future Internet* **2022**, *14*, 260. https://doi.org/10.3390/fi14090260

Academic Editors: Filipe Portela and Paolo Bellavista

Received: 7 July 2022
Accepted: 31 August 2022
Published: 31 August 2022

1. Introduction

In recent years, deep learning methods have achieved great success in solving various problems, including bioinformatics [1], cybersecurity [2], manufacturing [3,4], or natural language processing (NLP). NLP deals with creating computational algorithms for the automatic analysis and representation of human language. In the field of NLP, neural networks achieve excellent results compared to traditional machine learning models, such as SVM (support vector machine) or logistic regression. In comparison to traditional machine learning algorithms, deep learning algorithms can learn multiple levels of representation. From the perspective of NLP, deep learning models (especially recursive neural networks) can also capture sequence information within the text (e.g., phrases), which makes them a more suitable option for NLP than the traditional methods. In recent years, convolutional neural networks (CNNs) have shown breakthroughs in some NLP tasks, such as text classification [5–7].

Nowadays, online platforms are a widespread phenomenon that enables users to communicate with different messages. Moving human communication to online platforms is a

double-edged sword. Benefits include the opportunity to share opinions and experiences and get immediate feedback, as well as the opportunity to discuss various topics. On the other hand, on these online platforms, we can observe vulgarities, hate speech, insults, or misinformation, which are referred to as *antisocial behavior on the Internet* [8]. Spreading misinformation on the Internet can take various forms, such as hoaxes, spam, rumors, false reviews, etc. We focused on two types of them, toxicity in comments and fake news [9]. Toxicity in comments is defined as a rude, disrespectful, or inappropriate comment that is likely to force other users to leave the discussion. Toxicity in comments can appear in various areas, such as social networks or discussions related to news articles [10]. The most common type of antisocial behavior on the Internet is fake news. They are considered news pieces that are intentionally and demonstrably untrue. Usually, these articles are designed to mislead, deceive, and influence people's opinions. Fake news contains false information, the veracity of which can be verified [11].

Manually detecting and tracking online content is a very demanding and costly process. Machine-learning systems that prescreen content and identify suspicious cases have proven successful in detecting antisocial behavior. These algorithms may prove to be a viable solution to problems on social networking platforms.

Besides the traditional machine learning models, deep learning is very capable in the detection of various forms of antisocial behavior on the web. Deep networks (including different topologies of CNN) have been successfully used to automatically detect cyber-bullying in Twitter posts [12,13]. Deep networks are successful in other related tasks, such as hate speech detection [14]. Besides the commonly used deep learning architectures, ensembles of deep networks can be used to improve the detection ratio [15]. Deep learning methods are very popular in toxic comment classification and fake news detection. The authors of the study [16] focused on the detection of fake news using neural network methods on two datasets that contained English news articles. To solve this problem, the authors used CNNs, RNNs (recurrent neural networks), unidirectional LSTM (long short-term memory), and bidirectional LSTM networks. In [17], the authors focused on binary toxicity classification in online comments. The authors used the k-nearest neighbors, naive Bayes, and CNN models to detect toxicity in the comments. The CNN network proved to be the most successful one. In contrast to the previous study, the authors of [18] focused on classifying toxicity in comments and minimizing identity bias. In this experiment, the authors showed that although the model works well on a dataset, it can still demonstrate bias at subgroup levels. In this experiment, they trained three models: LSTM, BERT, and the TF-IDF model. As in previous studies, the authors [19] focused on detecting toxicity in online comments. The authors divided this research into two parts. In the first part, they used a binary classification to detect toxic comments correctly. In the second part, they used a multiclass classification model to determine the degree of toxicity. Deep networks can address different types of toxicity in the text using multilabel classification [20,21]. Capsule networks are also used to track the temporal aspects of toxicity in the comments [22].

The use of ML models to detect antisocial behavior from the texts is well studied, and models based on neural networks often prove to be the most suitable for handling these tasks. However, there are still many open research issues. One of the problems lies in the lack of well-labeled data. Even if many public datasets are available (in multiple antisocial behavior detection areas), many of them are still human-labeled, which may incorporate bias into the data. On the other hand, automatic labeling (e.g., using lexicons) may be more efficient in processing more data but still is not very reliable. To address the bias which can be introduced by human labeling, techniques such as crowd sourcing can be utilized. In addition, many datasets in this domain are heavily imbalanced. Such class imbalance may influence the detection models' performance, especially in the minor class, which usually represents the type of antisocial behavior (e.g., fake news articles, fake reviews, or toxic comments). Therefore, exploring the approaches that can sample the data in the minor

classes to overcome the lack of data can be interesting. An important issue is generating new, artificial samples with the same characteristics as the original data.

In work presented in this paper, we focused on using data augmentation techniques to improve the class imbalance by generating new, artificial samples from minor classes. We used simple text transformation methods based on vocabularies which were used to construct the new samples by replacing certain words with their synonyms. Such methods were already experimentally evaluated in several domains, e.g., clinical literature [23], sentiment analysis [24], or more recently in local (Portuguese) fake news detection [24]. The main motivation of our research was to focus on the antisocial detection behavior domain. We selected two typical tasks within this area which often involve processing imbalanced data. Then we experimentally evaluated if the application of EDA in these tasks could influence the performance of the detection models. We decided to evaluate both, separate EDA techniques as well as a combination of all EDA methods applied at once. We compared the performance of the classification models on the original and EDA-extended datasets.

The paper is organized as follows: Section 2 describes the data augmentation methods used in the text processing domain. The following section presents the datasets used in the study and their preprocessing. Section 4 presents the deep learning model used in both evaluated tasks and presents the results of the experiments. Section 5 presents the conclusions of the experiment's results.

2. Data Augmentation

In deep learning, model performance often improves with increasing data volume. To achieve the required performance of the model, it is necessary to have enough training data, which, however, are not always available. To solve this problem, it is essential to add new data samples or generate new samples from existing ones to be able to train a more generalized model. Manually adding relevant new records is a very time-consuming process. This has led to the development of methods to generate new samples automatically. *Augmentation techniques* are used to automatically extend the size of the dataset used to train the models. Data augmentation generally involves methods that increase the training data without collecting new data. There are multiple data augmentation techniques; most are based on a generation of slightly perturbed original data samples. The main motivation is to use the augmented data as a regularizer to reduce the potential overfitting of the trained models. Therefore, generated data samples should be neither too different (semantically) nor, at the same time, too similar to the original samples. This could lead to an even higher level of overfitting. We can use these techniques to handle two problems that may cause poor model performance. The first one is the lack of data in the training set. The second problem lies in the uneven distribution of records in the target class. In the case of heavy imbalanced classes, the model usually performs better in a major class, while it struggles to learn concepts from the minor one. Data augmentation techniques can help to artificially expand the size of a training set by creating new records from existing data (usually from the minor classes) [25].

2.1. Data Augmentation in NLP

While using augmentation techniques in computer vision and image classification applications is very popular [26–29], text data augmentation techniques in NLP applications are still relatively less used mostly due to the complexity of natural language. However, more recently, methods to extend textual training data for NLP tasks have been the subject of several studies [30–32]. In general, augmentation methods in NLP can be divided into four groups based on the level at which the augmentation procedure works. Character-level augmentation is based on adding noise to the training samples (e.g., switching the letters) on a character level. Such an approach can be used to simulate the natural noise in text (e.g., spelling mistakes), common in written text. Models trained on character-level enhanced data should be more suitable for text processing tasks in some specific

domains, e.g., classification of texts from social networks [33,34]. Word-level augmentation is based on adding noise or the replacement of words. Very common are vocabulary-based approaches, where replacements are based on different lexicons [35–37]. One of the most frequently used ones is Wordnet [38]. A very popular word-level augmentation technique is easy data augmentation (EDA) [39], which we will describe later. Besides the replacement of the lexical units, there are more advanced methods based on embedding replacement which (instead of synonym substitution) replace words based on textual context [40] or replacement methods based on language models [41]. Phrase- or sentence-level augmentation methods perform the replacement of multiple words or parts of the text (e.g., semantic text exchange [42]). In the last group, document-level methods are used to generate entire text documents. In this case, mostly generative approaches[43] or translation models [44,45] are used.

2.2. Easy Data Augmentation

EDA consists of augmentation techniques developed and evaluated in [39]. This set of methods uses traditional and straightforward approaches to increase the volume of data. These techniques create new data so that the meaning and grammatical structure of the original data are preserved. EDA consists of the following four methods that prevent model learning and help train more robust models:

- **Synonyms replacement**—this method creates new samples of textual documents by the replacement of suitable words from the text with their synonyms which retain the meaning of the text. The user needs to provide the parameter *n*, which specifies how many words will be replaced by their synonyms in the given sentence. Users must also focus on selecting the right words and appropriate synonyms that do not change the meaning of the text. In our research, we used the Wordnet lexical database for the English language, which is available in the NLTK library in Python. In addition, we used the Stopwords dictionary (a corpus containing English stop words) to identify nonmeaningful words in the data. An example of this method on the toxic comments dataset used in the study is as follows:
  All *stupid, no brains, is a more* ***accurate*** *description of Trudeau.*
  Wholly *stupid, no brains, is a more* ***precise*** *description of Trudeau.*
- **Random insertion**—this method works with synonyms of selected words as in the previous method. However, the difference is that *n* synonyms of the words chosen are randomly inserted at different positions in the text. The random insertion of the synonyms of certain words from the text, as opposed to a synonyms replacement, can be more relevant to a given context, as it retains the original evaluation of the text. The method is specified by the number of full-meaning words whose synonyms are randomly inserted at particular positions. An example of an application of this method on a sample record from the toxic comments database is as follows:
  Presidents *are unpredictable. What do you* ***think*** *we should do about it, shoot him in the* ***back?***
  Presidents are unpredictable. What ***recover*** *do you think we should do* ***chairperson retrieve*** *about it, shoot him in the back?*
- **Random swap**—unlike previous methods, this method does not use synonyms to create new samples. This method aims to select two random words that swap positions in the text. The user specifies parameter *n*, which defines how many pairs of words should exchange positions in the text. This method could help the model to be more robust. However, a very high number of words exchanged can worsen the model due to a change in the meaning of the text. An illustrative example of this method on the toxic comments dataset is as follows:
  Only ***wishful*** *thinking on* ***your part****—simple and silly* ***imagining****. something would happen that* ***isn't realistic****.*
  Only ***your*** *thinking on* ***wishful isn't*** *—simple and silly* ***realistic****. something would happen that* ***part imagining****.*

- **Random deletion**—the goal of this method is to randomly remove words in the text with a certain probability specified by a parameter p. It is necessary to choose the probability appropriately so we do not delete many words from the text. The text in the toxic comments dataset after applying this augmentation technique is changed as follows:
  *Funny **you** should link Palin and Trump. They are both grifters, **playing** the poorly educated for fools.*
  Funny should link Palin and Trump. They are both grifters, the poorly educated for fools.

The EDA provides simple methods for increasing the training data volume, but it is important to follow certain rules for the methods to work properly. In each method, it is necessary to correctly choose the number of words to be replaced, inserted, deleted, or replaced so that the meaning and structure of the text are preserved. For text with a small number of words, the meaning of the text is more likely to change. It is also very important that methods replace only full-meaning words and not stop words. Properly tuning the method parameters does not change the text's meaning or structure, and the model improves.

3. Data Understanding and Preprocessing

To evaluate the selected EDA methods, we used two datasets from the antisocial behavior detection domain—toxic comments and fake news datasets. In the first case, we used the Jigsaw toxic comments dataset (available online: www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification/data, accessed on 6 July 2022), which is used to train the models for the detection of toxicity in comments related to news reports. The dataset consists of short text comments that contain very informal language and expressions, including emoticons, explicit language, or slang expressions. The dataset was created by the Civil Comments platform, which collected and made available in competition Jigsaw Unintended Bias in Classification Toxicity. Figure 1 depicts a dataset sample to illustrate the content of the texts. Individual documents have lengths ranging from 1 to 1000 characters. The majority of the comments are short, ranging from 50 to 150 characters. The target attribute represents the toxicity score. The score values range from 0.0 to 1.0 and represent the fraction of raters who believed the label fit the toxicity type. We transformed the numeric target feature to a binary class, dividing the comments into toxic and nontoxic groups, where toxic comments were considered those with scores higher than 0.5. Other comments we considered as nontoxic/neutral. The resulting binary target feature was unbalanced, as the toxic comments made up only 8 percent of the entire dataset.

In the second task, we solved the fake news detection problem in news articles. The dataset consists of news articles; texts are usually longer than the comments used in the previous dataset. In addition, as the data consist of news pieces from different online media, the texts are written more formally and use more polished language. The particular documents are also longer, with the majority of the documents containing from 2000 to 5000 characters. The dataset consists of a total of 7500 news articles. The target class is binary, specifying if a given article is considered a regular news piece or if it contains misinformation. In addition, in this case, the target class is unbalanced, with regular records being a major and fake news a minor class.

In the data preprocessing phase, we performed just the basic standard text preprocessing, including converting the texts to lowercase, removing the punctuation marks, and dividing the sentences into tokens. We did not apply more text processing techniques to keep the data in form as close to original as possible. Data in datasets describing antisocial behavior contain various slang words, abbreviations, emoticons, or other forms of text that can express the features and characteristics of antisocial behavior. By removing these words, we could disrupt the main features and characteristics of the text, which could change the semantic meaning [46]. Especially with toxic comments, it is very important to keep the text form as close to the original to extract the features of toxicity in the comments properly. For the same reason, keeping the stop words, nonmeaningful words, and formu-

las is important. In the antisocial behavior detection domain, it is more suitable to keep the dataset without several more advanced preprocessing methods (such as stemming or lemmatization or stop words removal). The application of those techniques can result in a loss of important information typical for the style used in the short texts (comments) [6]. As a text representation model, we used GloVe (Global Vectors for Word Representation) embeddings [47]. We aligned the sequences to the same length. In the toxic comments dataset, we used a maximum size of 200. In the fake news dataset, we set a maximum size of 2500.

Index	comment_text	target
0	This is so cool. It's like, 'would you want your mother to read this??' Really great idea, well done!	0
1	Thank you!! This would make my life a lot less anxiety-inducing. Keep it up, and don't let anyone get in your way!	0
2	This is such an urgent design problem; kudos to you for taking it on. Very impressive!	0
3	Is this something I'll be able to install on my site? When will you be releasing it?	0
4	haha you guys are a bunch of losers.	0.893617
5	ur a sh*tty comment.	0.666667
6	hahahahahahahahhha suck it.	0.457627
7	FFFFUUUUUUUUUUUUUU	0
8	The ranchers seem motivated by mostly by greed; no one should have the right to allow their animals destroy public land.	0
9	It was a great show. Not a combo I'd of expected to be good together but it was.	0
10	Wow, that sounds great.	0
11	This is a great story. Man. I wonder if the person who yelled "shut the fuck up!" at him ever heard it.	0.44
12	This seems like a step in the right direction.	0
13	It's ridiculous that these guys are being called "protesters". Being armed is a threat of violence, which makes them terrorists.	0.6
14	This story gets more ridiculous by the hour! And, I love that people are sending these guys …	0.5
15	I agree; I don't want to grant them the legitimacy of protestors. They're greedy, small-mind…	0
16	Interesting. I'll be curious to see how this works out. I often refrain from commenting beca…	0

Figure 1. Target class frequency in the original and EDA-extended toxic comments dataset.

While applying the selected EDA techniques, we extended the dataset by newly created artificial samples from the minor class. In this phase, it was necessary to correctly choose the parameters of the EDA methods—the number of words to be replaced, inserted, exchanged, or deleted. We trained multiple CNN models to find out which parameter value would be optimal and recorded the best results. On the toxic comments dataset, we decided to use the parameters $n = 4$ in synonyms replacement, $n = 3$ in random insertion and random swap, and $p = 0.13$ in the random deletion approach. Similarly, on the fake news dataset, we applied the EDA techniques using following settings: synonyms replacement ($n = 4$), random insertion ($n = 6$), random swap ($n = 3$), and random deletion ($p = 0.12$). First, we gradually added individual techniques and, finally, we applied a combination of all EDA methods. The application of particular EDA techniques doubled the minor class records in the training data; the application of all EDA techniques resulted in five times more samples of the minor class. Tables 1 and 2 then summarize the class attributes in both of the datasets before and after the application of EDA methods.

Table 1. Target class frequency in the original and EDA-extended toxic comments dataset.

Data Augmentation	Neutral Comments	Toxic Comments
Original	1,328,643	115,256
Synonym replacement	1,328,643	230,512
Random insertion	1,328,643	230,512
Random swap	1,328,643	230,512
Random deletion	1,328,643	230,512
EDA	1,328,643	576,280
Test set	331,897	29,078

Table 2. Target class frequency in the original and EDA-extended fake news dataset.

Data Augmentation	Relevant News	Fake News
Original	5618	460
Synonym replacement	5618	920
Random insertion	5618	920
Random swap	5618	920
Random deletion	5618	920
EDA	5618	2300
Test set	1382	138

4. Detection of Antisocial Behavior Using Deep Learning Methods

We chose a CNN [48] model for the experiments, as the architecture proved to achieve good results in NLP tasks [13,49]. While maintaining performance, CNN architecture proved to be much less computationally intensive than LSTM networks. We expected that the effect of EDA augmentations should be very similar regardless of the used model. Both solved problems were binary classification tasks. Binary classification aims to classify data into one of two classes. In our case, we classified the data in the first dataset into toxic/nontoxic comments and, in the case of the fake news dataset, into fake/relevant news. Entire preprocessing, training, and evaluation were implemented in the Python language, including standard analytical stack (e.g., Pandas, Tensorflow, and scikit-learn packages).

The main idea of the experiments was to find out the effect of EDA augmentation techniques to the classification results. We used a simple convolutional neural network model, shown in Figure 2. It consisted of two convolution layers, two pooling layers, one flatten layer, and one regularization dropout layer. We used the checkpoint method to prevent overfitting [50]. The hyperparameters of the CNN model are summarized in Table 3. During the experiments, we gradually added individual EDA augmentation techniques to the data and monitored its influence on the resulting metrics.

We evaluated the models using standard classification metrics:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

$$\text{F1 score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}. \tag{4}$$

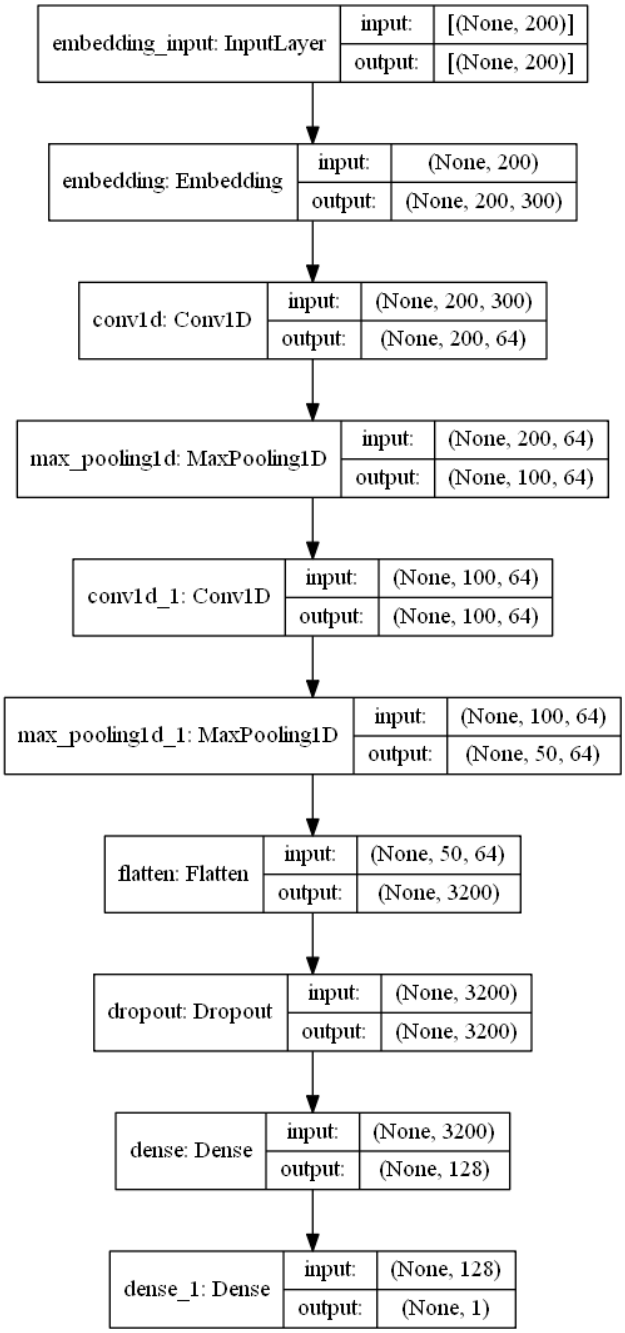

Figure 2. Architecture of CNN model.

Table 3. Hyperparameters of the CNN model.

Hyperparameters	Values
Batch size	32
Optimizer	Adam
Learning rate	0.001
Dropout rate	0.20

These metrics were computed using the coefficients derived from the confusion matrix (see Table 4), which expresses the number of correct and incorrect predictions made by the classification model compared to the ground truth values in the testing data. In the formulas, TP, FP, FN, and FP stand for true positive, false positive, false negative, and false positive rates associated with the class attribute. These metrics were used also to measure

the model performance on the particular minor. To measure the overall model performance, we also used the AUC (area under curve). The AUC score computes the area under the ROC (receiver operating characteristic) curve and provides the aggregate measure of model performance across all possible classification thresholds.

Table 4. Confusion matrix.

		Actual Values	
		0	**1**
Predicted Values	**0**	TN	FN
	1	FP	TP

4.1. Evaluation of Toxic Comments Detection

The basic model of the convolutional neural network without the extension of the training set reached a precision of 78%, a recall of 50%, and an F1 score of 61% in toxicity detection (minor class prediction, see Table 5). After evaluating the basic model, we trained the models using EDA augmentation techniques. Table 6 summarizes the overall performance (macro-averaged metrics) of the CNN model and compares different EDA methods applied to the original training data.

Table 5. Performance results of the CNN model on the toxic comments dataset achieved in the minor (toxic) class.

Data	Precision	Recall	F1 Score
Original	0.78	0.50	0.61
Synonym replacement, n = 4	0.74	0.55	0.63
Random insertion, n = 3	0.74	0.55	0.63
Random swap, n = 3	0.76	0.52	0.62
Random deletion, p = 0.13	0.77	0.51	0.61
EDA	0.51	0.77	0.61

Table 6. Macro-averaged performance results of the CNN model on the toxic comments dataset.

Data	Accuracy	Precision	Recall	F1 Score	AUC
Original	0.95	0.87	0.74	0.79	0.9461
Synonym replacement, n = 4	0.95	0.85	0.76	0.80	0.9509
Random insertion, n = 3	0.95	0.85	0.77	0.80	0.9508
Random swap, n = 3	0.95	0.86	0.75	0.80	0.9505
Random deletion, p = 0.13	0.95	0.86	0.75	0.79	0.9505
EDA	0.92	0.74	0.85	0.78	0.9422

In this case, EDA augmentation techniques did not achieve the desired improvements in the detection of toxicity in the comments. Although we trained models with different augmentation techniques where we defined other parameters, we still did not achieve significant improvement. The toxic comments dataset contains comments with many slang words, dialect words, abbreviations, swear words, and words made up of different characters. For this reason, augmentation techniques that work with synonyms cannot be used because they are not in the standard synonym dictionary. If these words are replaced by words that do not constitute antisocial behavior, false negative cases will arise. When randomly deleting, the biggest problem is the length of the text. As these are short texts, toxic words are often deleted. After deleting these words, the comment becomes neutral, so there arise false negative cases.

4.2. Evaluation on the Fake News Dataset

The CNN model was trained on the fake news dataset without the extension of the training data, with an unbalanced target attribute class. The training set consisted of more than 92% regular news pieces and only 8% fake news articles. After training the base model of the same architecture as in the previous dataset, we evaluated the model's performance on the minor (fake news) class. The fake news dataset contained 460 positive cases in the training set. Table 7 shows the confusion matrix of this model.

Table 7. Confusion matrix of base CNN model trained on fake news dataset.

		Actual Values	
		0	**1**
Predicted Values	**0**	1378	4
	1	66	72

As in the previous dataset, using EDA augmentation techniques extended the data with approximately more than four times more fake news records. Table 8 summarizes the results of the CNN model on the original data and data extended using EDA techniques. The CNN model achieved a precision of 95%, a recall of 52%, and an F1 score of 67% in the toxic comments class. Table 9 summarizes the macro-averaged performance of the CNN model on the original and EDA-extended datasets.

Table 8. Performance results of the CNN model on the fake news dataset achieved in the minor (fake news) class.

Data	Precision	Recall	F1 Score
Original	0.95	0.52	0.67
Synonym replacement, n = 6	0.91	0.68	0.78
Random insertion, n = 6	0.90	0.63	0.74
Random swap, n = 3	0.96	0.64	0.77
Random deletion, p = 0.12	0.93	0.63	0.75
EDA	**0.95**	**0.79**	**0.86**

Table 9. Macro-averaged performance results of the CNN model on the fake news dataset.

Data	Accuracy	Precision	Recall	F1 Score	AUC
Original	0.95	0.95	0.76	0.82	0.9701
Synonym replacement, n = 6	0.97	0.94	0.84	0.88	0.9781
Random insertion, n = 6	0.96	0.93	0.81	0.86	0.9773
Random swap, n = 3	0.97	0.96	0.82	0.88	0.9785
Random deletion, p = 0.12	0.96	0.94	0.81	0.86	0.9758
EDA	**0.98**	**0.96**	**0.89**	**0.92**	**0.9856**

The model, which used a combination of all EDA augmentation methods, significantly improved all metrics compared to other models. Using EDA, we increased the F1 score by 19 percent and recall by 27 percent when detecting fake news samples (minor class). The confusion matrix of this model is shown in Table 10.

Table 10. Confusion matrix of CNN model trained on the fake news dataset expanded using EDA techniques.

		Actual Values	
		0	**1**
Predicted Values	**0**	1376	6
	1	29	109

4.3. Comparison with Related Literature

The use of EDA techniques was already explored in the available literature [39], where the authors evaluated these methods on five different text classification tasks. EDA boosted the model's performance marginally, but significant improvements could be expected on the smaller datasets. Similar behavior was observed during our experiments. Especially on the fake news dataset, which consisted of approximately 6000 training samples, the improvements were most significant. To check the issue of the possible overfitting that EDA may cause, we evaluated the models' overall performance and their performance in the minor class. This kind of evaluation supported the benefits that EDA applications can bring.

In addition, we could compare the results obtained by applying similar techniques to the same dataset. For example, in [51], authors applied EDA and back-translation techniques to the toxic comments classifiers using traditional machine learning algorithms (logistic regression and support vector machine). Their baseline model gained an F1 performance of 0.677; after the EDA application, it improved to 0.736. Back-translation itself did not achieve better results. Relatively lower levels of F1 could be attributed to the usage of standard ML models. In addition, in this case, the authors did not perform the optimization of EDA parameters or their combination. In [19], authors used a similar technique as EDA on CNN-based toxic comments classification. The authors used synonym replacement, random mask, and unique word augmentations, which improved the baseline CNN model from a 0.846 F1 to 0.885 (a combination of all techniques). Similar to our experiments, a combination of multiple techniques brought higher benefits to the model performance. The difference in the F1 score of the baseline model can be attributed to different test sets (the testing set in our experiments consisted of 20% of samples, in comparison to a 10% test set in these experiments).

The comparison of the fake news dataset can be difficult, as there are multiple datasets, and their usage among the studies is rather inconsistent. The majority of current research [52–54] uses the COVID-19 fake news dataset. However, mentioned studies use advanced deep learning models for classification and different augmentation techniques (translation, BiGRU-CRF, and CapsuleNet). In both cases, the effects of the applied techniques are quite similar, as they boost the F1 performance of the classifiers by 0.01. It is important to note that applying more advanced techniques can be very demanding on computational resources. EDA techniques can be relatively simple to implement, but their effects on classifier performance can be comparable.

4.4. Practical Implications

In this work, we have presented a CNN classification model for the detection of two forms of antisocial behavior detection trained on an EDA-extended dataset. Data analytical methodologies such as CRISP-DM [55] describe the overall data analysis process in multiple steps. Such steps include data understanding and preparation, the training and evaluation of the analytical models, and the actual deployment of the model into production. In this paper, we focused mostly on the experimental evaluation of the model. However, as we used standard data analytical technologies in the implementation, it is relatively straightforward to serialize the developed models to transfer them into the production environment. In the studied domain, similar models could run as web services, consuming the input data from the sources and providing real-time predictions. Depending on the particular type of task, such models could be implemented as browser extensions highlighting the given text (e.g., toxic comments or unreliable news pieces) during web browsing. From a practical point of view, data can be accessed in real-time using public APIs (e.g., news articles or comments from social networks). Models can be serialized using standard Python tools (e.g., Pickle) and deployed as web services using a web framework (e.g., Flask). Such an approach enables the creation of an architecture where serialized models are used to score the incoming data on the back-end and feed the classification

results to the front-end. In this case, the output of the model can be fed to the browser extension, able to highlight possible toxic comments or unreliable news pieces.

5. Discussion and Conclusions

The work presented in this paper focused on data augmentation techniques applied to text classification in the antisocial behavior detection domain. The main objective was to explore the possibility of using simple EDA augmentation techniques to overcome the class imbalance problem when solving antisocial behavior detection tasks using deep learning models. We evaluated EDA methods on two selected tasks—fake news detection and toxic comments classification. In both cases, we used the CNN classifier and compared its performance when trained on the original training set with training sets enhanced using a combination of EDA techniques. The effect of EDA augmentation techniques on the model performance is strongly dependent on the dataset. Although there are multiple EDA techniques available, those are usually very well used when applied to a dataset containing the texts written in more formal language. It was evident on the performance boost of the CNN model on the fake news dataset, which was significant, improving the F1 score by a 0.1.

From the perspective of the style and language of the texts, EDA techniques applied to the fake news dataset positively affected classification performance. This dataset comprised news pieces usually longer than discussion comments and written using more formal language. This task was much better suited to the EDA synonym replacements and similar techniques. Using EDA, we could correctly generate the augmented data samples, contributing to model performance improvement. On the other hand, EDA techniques applied to the dataset of toxic comments did not improve the CNN model performance (only a 0.01 improvement in F1). The dataset mostly comprised short texts (discussion posts) and contained much nonformal content (e.g., slang expressions). Therefore, EDA methods relying on synonyms replacement were unable to find suitable synonyms for many of the words typical for toxic behavior in the comments. The application of these methods did not generate the augmented toxic samples suitable enough to be used to improve the model's performance.

In general, the problem of enhancing the datasets (e.g., due to data scarcity or to balance the classes) in the NLP domain is very difficult and attracts the attention of many research groups. The ability to artificially generate new texts is a difficult task, and it is very challenging to synthetically generate the features present in the texts written by humans (such as irony or sarcasm). Simple augmentation techniques such as EDA cannot reflect these complex issues and even simpler ones, such as considering the context of replaced words. However, its simple implementation and application while maintaining reasonable performance can present an advantage in certain applications. In the future, we expect that exploration of the usage of a character-level augmentation method could be useful, as they can generate texts which can represent spelling mistakes (which are very common in this type of data). The further analysis and exploration of the suitability evaluation of other, more advanced augmentation methods such as GAN in this domain could be interesting.

Author Contributions: Conceptualization, M.S. and V.M.-K.; methodology, M.S.; software, V.M.-K. and J.J.; validation, V.M.-K.; formal analysis, M.S.; investigation, V.M.-K. and J.J.; resources, M.S.; writing—original draft preparation, V.M.-K. and M.S.; writing—review and editing, V.M.-K. and M.S.; supervision, M.S.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by VEGA grant number 1/0685/21.

Data Availability Statement: Not Applicable, the study does not report any data.

Acknowledgments: The studies of the deep learning approach presented in this work are supported by Slovak VEGA research grant No. 1/0685/21.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, Y.; Huang, C.; Ding, L.; Li, Z.; Pan, Y.; Gao, X. Deep learning in bioinformatics: Introduction, application, and perspective in the big data era. *Methods* **2019**, *166*, 4–21. [CrossRef] [PubMed]
2. Mahdavifar, S.; Ghorbani, A.A. Application of deep learning to cybersecurity: A survey. *Neurocomputing* **2019**, *347*, 149–176. [CrossRef]
3. Yang, J.; Li, S.; Wang, Z.; Dong, H.; Wang, J.; Tang, S. Using Deep Learning to Detect Defects in Manufacturing: A Comprehensive Survey and Current Challenges. *Materials* **2020**, *13*, 5755. [CrossRef] [PubMed]
4. Ferencek, A.; Kofjač, D.; Škraba, A.; Sašek, B.; Borštnar, M.K. Deep Learning Predictive Models for Terminal Call Rate Prediction during the Warranty Period. *Bus. Syst. Res. J.* **2020**, *11*, 36–50. [CrossRef]
5. Risch, J.; Krestel, R. *Toxic Comment Detection in Online Discussions*; 2020. Available online: https://link.springer.com/chapter/10.1007/978-981-15-1216-2_4 (accessed on 6 July 2022). CrossRef]
6. Maslej-Krešňáková, V.; Sarnovský, M.; Butka, P.; Machová, K. Comparison of Deep Learning Models and Various Text preProcessing Techniques for the Toxic Comments Classification. *Appl. Sci.* **2020**, *10*, 8631. [CrossRef]
7. Sarnovský, M.; Butka, P.; Bednár, P.; Babič, F.; Paralič, J. Analytical Platform Based on Jbowl Library Providing Text-Mining Services in Distributed Environment. In *Proceedings of the Information and Communication Technology*; Khalil, I., Neuhold, E., Tjoa, A.M., Xu, L.D., You, I., Eds.; Springer International Publishing: Cham, Switzerland, 2015; pp. 310–319.
8. Burney, E. *Making People Behave: Anti-Social Behaviour, Politics and Policy, 2nd ed.*; 2013 . Available online: https://www.taylorfrancis.com/books/mono/10.4324/9781843927112/making-people-behave-elizabeth-burney (accessed on 6 July 2022).
9. Cheng, J.; Danescu-Niculescu-Mizil, C.; Leskovec, J. Antisocial behavior in online discussion communities. In Proceedings of the 9th International Conference on Web and Social Media, ICWSM, Oxford, UK, 26–29 May 2015 .
10. Machova, K.; Srba, I.; Sarnovský, M.; Paralič, J.; Kresnakova, V.M.; Hrckova, A.; Kompan, M.; Simko, M.; Blaho, R.; Chuda, D.; et al. Addressing False Information and Abusive Language in Digital Space Using Intelligent Approaches. In Proceedings of the World Symposium on Digital Intelligence for Systems and Machines, Prague, Czech Republic, 24–26 June 2020; pp. 3–32.
11. Shu, K.; Sliva, A.; Wang, S.; Tang, J.; Liu, H. Fake News Detection on Social Media: A Data Mining Perspective. *SIGKDD Explor. Newsl.* **2017**, *19*, 22–36. [CrossRef]
12. Anindyati, L.; Purwarianti, A.; Nursanti, A. Optimizing Deep Learning for Detection Cyberbullying Text in Indonesian Language. In Proceedings of the Proceedings—2019 International Conference on Advanced Informatics: Concepts, Theory, and Applications, ICAICTA 2019, Yogyakarta, Indonesia, 20–21 September 2019. [CrossRef]
13. Al-Ajlan, M.A.; Ykhlef, M. Deep Learning Algorithm for Cyberbullying Detection. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9* . Available online: https://thesai.org/Publications/ViewPaper?Volume=9&Issue=9&Code=ijacsa&SerialNo=27 (accessed on 6 July 2022).
14. Ranasinghe, T.; Zampieri, M.; Hettiarachchi, H. BRUMS at HASOC 2019: Deep learning models for multilingual hate speech and offensive language identification. In Proceedings of the CEUR Workshop Proceedings, Stuttgart, Germany, 19 February 2019.
15. Zimmerman, S.; Fox, C.; Kruschwitz, U. Improving hate speech detection with deep learning ensembles. In Proceedings of the LREC 2018—11th International Conference on Language Resources and Evaluation, Miyazaki, Japan, 7–12 May 2019.
16. Bahad, P.; Saxena, P.; Kamal, R. Fake News Detection using Bi-directional LSTM-Recurrent Neural Network. *Procedia Comput. Sci.* **2019**, *165*, 74–82. [CrossRef]
17. Georgakopoulos, S.V.; Tasoulis, S.; Vrahatis, A.G.; Plagianakos, V. Convolutional Neural Networks for Toxic Comment Classification. In Proceedings of the 10th Hellenic Conference on Artificial Intelligence, Patras Greece, 9–12 July 2018.
18. Ashod Zorian, A.; Shekar Bikkanur, C. Debiasing Personal Identities in Toxicity Classification. *arXiv* **2019**, arXiv:1908.05757.
19. Ibrahim, M.; Torki, M.; El-Makky, N. Imbalanced Toxic Comments Classification Using Data Augmentation and Deep Learning. In Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 17–20. December 2018; pp. 875–878. [CrossRef]
20. Krešňáková, V.M.; Sarnovský, M.; Butka, P. Deep learning methods for Fake News detection. In Proceedings of the 2019 IEEE 19th International Symposium on Computational Intelligence and Informatics and 7th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics (CINTI-MACRo), Szeged, Hungary, 14–16 November 2019; pp. 143–148.
21. Mestry, S.; Singh, H.; Chauhan, R.; Bisht, V.; Tiwari, K. Automation in Social Networking Comments with the Help of Robust fastText and CNN. In Proceedings of the 1st International Conference on Innovations in Information and Communication Technology, ICIICT 2019, Chennai, India, 25–26 April 2019. [CrossRef]
22. Srivastava, S.; Khurana, P.; Tewari, V. Identifying Aggression and Toxicity in Comments using Capsule Network. In Proceedings of the COLING 2018—1st Workshop on Trolling, Aggression and Cyberbullying, TRAC 2018—Proceedings of the Workshop, Santa Fe, NM, USA, 20–21 August 2018.
23. Kang, T.; Perotte, A.; Tang, Y.; Ta, C.; Weng, C. UMLS-based data augmentation for natural language processing of clinical research literature. *J. Am. Med. Inform. Assoc.* **2020**, *28*, 812–823. [https://academic.oup.com/jamia/article-pdf/28/4/812/36642182/ocaa309.pdf] (accessed on 6 July 2022). [CrossRef] [PubMed]
24. Abonizio, H.Q.; Paraiso, E.C.; Barbon Junior, S. Toward Text Data Augmentation for Sentiment Analysis. *IEEE Trans. Artif. Intell.* **2021**, 1 . Available online: https://ieeexplore.ieee.org/document/9543519 (accessed on 6 July 2022).

25. Badimala, P.; Mishra, C.; Modam Venkataramana, R.K.; Bukhari, S.; Dengel, A. A Study of Various Text Augmentation Techniques for Relation Classification in Free Text. 2019; pp. 360–367. Available online: https://www.scitepress.org/Link.aspx?doi=10.5220/0007311003600367 (accessed on 6 July 2022).
26. Perez, L.; Wang, J. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. *arXiv* **2017**, [arXiv:cs.CV/1712.04621].
27. Fawzi, A.; Samulowitz, H.; Turaga, D.; Frossard, P. Adaptive data augmentation for image classification. In Proceedings of the 2016 IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3688–3692. [CrossRef]
28. Mikołajczyk, A.; Grochowski, M. Data augmentation for improving deep learning in image classification problem. In Proceedings of the 2018 International Interdisciplinary PhD Workshop (IIPhDW), Swinoujscie, Poland, 9–12 May 2018; pp. 117–122. [CrossRef]
29. Shorten, C.; Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 1–48. [CrossRef]
30. Shorten, C.; Khoshgoftaar, T.M.; Furht, B. Text Data Augmentation for Deep Learning. *J. Big Data* **2021**, *8*, 1–34. [CrossRef] [PubMed]
31. Bayer, M.; Kaufhold, M.A.; Reuter, C. A Survey on Data Augmentation for Text Classification. *ACM Comput. Surv.* **2022** . Available online: https://dl.acm.org/doi/10.1145/3544558 (accessed on 6 July 2022).
32. Feng, S.Y.; Gangal, V.; Wei, J.; Chandar, S.; Vosoughi, S.; Mitamura, T.; Hovy, E. A Survey of Data Augmentation Approaches for NLP. *arXiv* **2021**, arXiv:2105.03075 .
33. Belinkov, Y.; Bisk, Y. Synthetic and Natural Noise Both Break Neural Machine Translation. *arXiv* **2017**, arXiv:1711.02173.
34. Coulombe, C. Text Data Augmentation Made Simple By Leveraging NLP Cloud APIs. *arXiv* **2018**, arXiv:1812.04718.
35. Marivate, V.; Sefara, T. Improving Short Text Classification Through Global Augmentation Methods. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Glasgow, UK, 23–28 August 2020. [CrossRef]
36. Qiu, S.; Xu, B.; Zhang, J.; Wang, Y.; Shen, X.; De Melo, G.; Long, C.; Li, X. EasyAug: An Automatic Textual Data Augmentation Platform for Classification Tasks. In Proceedings of the The Web Conference 2020—Companion of the World Wide Web Conference, WWW 2020, Taipei Taiwan, 20–24 April 2020. [CrossRef]
37. Kobayashi, S. Contextual augmentation: Data augmentation bywords with paradigmatic relations. In Proceedings of the NAACL HLT 2018—2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies—Proceedings of the Conference, New Orleans, LA, USA, 1–6 June 2018. [CrossRef]
38. Miller, G.A.; Beckwith, R.; Fellbaum, C.; Gross, D.; Miller, K.J. Introduction to wordnet: An on-line lexical database. *Int. J. Lexicogr.* **1990** . Available online: https://academic.oup.com/ijl/article-abstract/3/4/235/923280?redirectedFrom=fulltext (accessed on 6 July 2022).
39. Wei, J.; Zou, K. 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)*, Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Hong Kong, China, 2019; pp. 6383–6389.
40. Wang, W.Y.; Yang, D. 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 the Conference Proceedings—EMNLP 2015: Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015. [CrossRef]
41. Wu, X.; Lv, S.; Zang, L.; Han, J.; Hu, S. Conditional BERT Contextual Augmentation. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Tours, France, 19–21 June 2020. [CrossRef]
42. Feng, S.Y.; Li, A.W.; Hoey, J. Keep calm and switch on! Preserving sentiment and fluency in semantic text exchange. In Proceedings of the EMNLP-IJCNLP 2019—2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference, Hong Kong, China, 3–7 November 2019. [CrossRef]
43. Sun, X.; He, J. A novel approach to generate a large scale of supervised data for short text sentiment analysis. *Multimed. Tools Appl.* **2020** . Available online: https://link.springer.com/article/10.1007/s11042-018-5748-4 (accessed on 6 July 2022).
44. Britz, D.; Goldie, A.; Luong, M.T.; Le, Q.V. Massive exploration of neural machine translation architectures. In Proceedings of the EMNLP 2017—Conference on Empirical Methods in Natural Language Processing, Proceedings, Copenhagen, Denmark, 9–11 September 2017. [CrossRef]
45. Kohli, H. Transfer Learning and Augmentation for Word Sense Disambiguation. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Hong Kong, China, 20–22 October 2021. [CrossRef]
46. Mohammad, F. Is preprocessing of text really worth your time for online comment classification? *arXiv* **2018**, arXiv:1806.02908.
47. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 26–28 October 2014; pp. 1532–1543.
48. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436. [CrossRef] [PubMed]
49. Georgakopoulos, S.V.; Vrahatis, A.G.; Tasoulis, S.K.; Plagianakos, V.P. Convolutional neural networks for toxic comment classification. In Proceedings of the ACM International Conference Proceeding Series, Tokyo, Japan 25–28 November 2018 . [CrossRef]

50. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
51. Rastogi, C.; Mofid, N.; Hsiao, F.I. Can We Achieve More with Less? Exploring Data Augmentation for Toxic Comment Classification. *arXiv* **2020**, arXiv:2007.00875.
52. Júnior, W.O.; da Cruz, M.S.; Wyzykowski, A.B.V.; de Jesus, A.B. The use of Data Augmentation as a technique for improving neural network accuracy in detecting fake news about COVID-19. *arXiv* **2022**, arXiv:2205.00452.
53. Karnyoto, A.S.; Sun, C.; Liu, B.; Wang, X. Augmentation and heterogeneous graph neural network for AAAI2021-COVID-19 fake news detection. *Int. J. Mach. Learn. Cybern.* **2022** . Available online: https://link.springer.com/article/10.1007/s13042-021-01503-5 (accessed on 6 July 2022).
54. Karnyoto, A.; Sun, C.; Liu, B.; Wang, X. Transfer learning and GRU-CRF augmentation for COVID-19 fake news detection. *Comput. Sci. Inf. Syst.* **2021** . Available online: http://www.doiserbia.nb.rs/Article.aspx?ID=1820-02142100053K (accessed on 6 July 2022).
55. Chapman, P.; Clinton, J.; Kerber, R.; Khabaza, T.; Reinartz, T.; Shearer, C.; Wirth, R. Crisp-Dm 1.0. *CRISP-DM Consort.* **2000**, 76 . Available online: https://ieeexplore.ieee.org/document/4579988/ (accessed on 6 July 2022).