Received: November 11, 2020. Revised: December 11, 2020.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Text Generation with Content and Structure-Based Preprocessing in Imbalanced Data of Product Review

Ana Alimatus Zaqiyah[1] **Diana Purwitasari[1]*** **Chastine Fatichah[1]**

[1]*Department of Informatics, Institut Teknologi Sepuluh Nopember, Indonesia*
* Corresponding author's Email: diana@if.its.ac.id

Abstract: Spam detection frequently categorizes product reviews as spam and non-spam. The spam reviews may contain texts of fake reviews and non-review statements describing unrelated things about products. Most of the publicly available spam reviews are labelled as fake reviews, while non-spam texts that are not fake reviews could contain non-review statements. It is crucial to notice those non-review statements since they convey misperception to consumers. Non-review statements are hardly found, and those statements of large and long texts often need to be manually labelled, which is time-consuming. Because of the rareness in finding non-review statements, there is an imbalanced condition between non-spam as a major class and spam that consists of the non-review statement as a minor class. Augmenting fake reviews to add spam texts is ineffective because they have similar content to non-spam such as some opinion words of product features. Thus, the text generation of non-review statements is preferable for adding spam texts. Some text generation issues are the frequent neural network-based methods require much learning data, and the existing pre-trained models produce texts with different contexts to non-review statements. The augmented texts should have similar content and context represented by the structure of the non-review statement. Therefore, we propose a text generation model with content and structure-based preprocessing to produce non-review statements, which is expected to overcome imbalanced data and give better spam detection results in product reviews. Structure-based preprocessing identifies the feature structures of non-opinion words from part-of-speech tags. Those features represent the context of spam reviews in unlabeled texts. Then, content-based preprocessing appoints selected topic modeling results of non-review statements from fake reviews. Our experiments resulted an improvement on the metric value of ± 0.04, called as BLEU (Bi-Lingual Evaluation Understudy) score, for the correspondence evaluation between generated and trained texts. The metric value indicates that the generated texts are not quite identical to the trained texts of non-review statements. However, those additional texts combined with the original spam texts gave better spam detection results with an increasing value of more than 40% on average recall score.

Keywords: Product review, Spam texts, Text generation, Topic modeling.

1. Introduction

Most consumers read product reviews as a consideration to buy a product [1]. The textual content in product reviews plays a significant role in controlling consumers' behavior and product demand variation [2]. This phenomenon influences some people to give untruthful opinions about the product to promote or damage their reputation. Some people also comment on unrelated things about the product, such as shipment, sellers, brands, advertising for other products, and questions. Thus, reviews that deceive people and non-review statements are categorized as spam. Previous works organize that fake reviews are called Type-1 spam, and non-review statements are called Type-3 spam [3]. Then, another texts called Type-2 spam emphasize on commenting brands. In this research, Type-2 are assumed as Type-3 because the texts are rarely found [4].

Most of the publicly available spam reviews are labeled as Type-1 spam [5, 6], while non-spam texts that are not Type-1 spam could contain Type-3 spam [3]. It is crucial to notice Type-3 spam since they convey misperception to consumers. These text examples are Type-3 spam, which are biased and could lead to incorrect judgment about the product:

- *"Excellent seller I got the cartridge on a timely manner ...",*
- *"I'm giving this a 2 stars only because my daughter also had this camera on her wish list."*
- *"the order was extremely difficult to complete online and the person on your phone help line did not a have clear grasp of the English language. That was unacceptable to me".*

Although those sample comments could be useful, they are not targeted at a specific product. Thus, text reviews could be marked as Type-3 spam if they have no opinion about a product and discuss non-related things about the product [3]. The existence of opinion on a product can be obtained by detecting product features' opinions in the sentence structure or the context sentences. The non-related thing about the product can be expressed from the content of the texts. Manually labeling Type-3 spam is time-consuming because the texts are up to more than 2-3 sentences. Because of rareness in finding Type-3 spam, there is an imbalance between non-spam as a major class and spam that consists of the Type-3 spam as a minor class.

Text generation helps to augment minor class to overcome imbalanced data. Augmenting Type-1 spam to add spam texts is ineffective because they have similar content and context to non-spam such as some opinion words of product features. Thus, text generation of Type-3 spam is preferable for adding spam texts. Two main approaches are used in text generation, statistical approach and deep-learning approach. Markov Chain introduced as the statistical approach of Latent Dirichlet Allocation (LDA) [7] models texts to topics based on word distribution. LDA depends on word distributions of contents and ignores word sequence-structures. The method could generate new texts, but unrealistic ones since they do not show logical meaning.

An encode-decode model with deep-learning, Recurrent Neural Network (RNN), was introduced as a solution to produce meaningful texts [8]. However, RNN has an exposure bias problem and the generated texts represent restricted word distribution of trained texts. Then, Generative Adversarial Network (GAN) was suggested as a deep-learning based generation model with concepts of discriminators and generators [9] to reduce exposure bias. However, GAN requires a large amount of data [10]. Insufficient data in text generation leads to produce unrealistic text, such as the model continually generates the same words.

Recently, Natural Language Processing (NLP) models are derived from Transformer model [11], which is the evolution of encoder-decoder module to trains faster and performs better. The Transformer decoder derives Generative Pre-Trained Transformer (GPT), a pre-trained model, to solve data insufficient problems. Pre-trained models could be less effective in specialized cases, such as the pre-trained models GPT [12] and GPT-2 [13] build texts with different content and context of Type-3 spam. The augmented texts should have the context of non-review statements marked by non-opinion words and the content of texts about unrelated things on the product. In our case, the pre-trained model needs fine-tuning step to learn how to represent the specialized texts of Type-3 spam which might unrecognizable. Text generation with fine-tuning step on trained data produces appropriate texts according to the content of non-related things and the context of non-opinion words.

In general, fine-tuning step requires more data that finally modifies word representation model. The following researches involve data addition to work on the word model such as texts for biomedical NLP [14], Autism Spectrum Disorder text-domain [15], and Malware Specific domain [16]. However, the data addition does not concern with text context and make their approaches are unfitting for augmentation Type-3 spam texts. Thus, this research focuses on utilizing the context and content based aspect to provide more augmented texts of Type-3 spam. The context could be represented as sentence structures.

Our research proposes a text generation model with content and structure-based preprocessing to produce non-review statements or known as Type-3 spam. Most of non-review statements contain non-opinion words. Thus, structure-based preprocessing would identify the part-of-speech tags of texts to obtain non-opinion words as a context representation of spam reviews in unlabeled texts. Another strong and focused point in our proposed model is to recognize non-review statements from fake reviews. Because fake reviews might have texts of non-review statement, the content-based preprocessing work on non-review statements to acquire the results of topic modeling. The strong point of our proposed model is to preserve the distribution and word structure of generated texts to have similar content and context with the original Type-3 spam. Therefore, the generated texts overcome the imbalanced problem by adding texts of Type-3 spam as the data of minor class to contribute better spam detection results.

This paper is organized as follows: a literature review of related works on overcoming imbalance data including recent text generation method such RNNs and Transformer model, also following researches involving data addition to work on the word model are written in Section 2. Our proposed text generation with content and structure-based preprocessing method is explained in Section 3. The experimental results and analysis of text generation in several existing methods and the impact of the proposed method in spam detection are discussed in Section 4. The conclusions consisting of the result of this work and future works are presented in Section 5.

2. Related works

Oversampling and under sampling are methods to handle imbalanced data. Augmentation is one of the oversampling methods that appends new data with synthetic data of the minor class. SMOTE is the most known technique proposed to generate synthetic data similar to the original one. AWH-SMOTE improved SMOTE with neighbors and noise identification to solve minority data that hardly represented in a dangerous region [17]. Nevertheless, SMOTE is not practical for high dimensional data like product reviews data. Another research also executes generating negative group data from the existing group data because manually recorded data only contained positive group data [18]. Generating new data can also be performed using text generation to augment minor class to overcome imbalanced data [19]. Statistical approaches such as Markov Chain [7] still fail to create realistic text and has logical meaning.

A deep learning method with an encode-decode (RNN) model is introduced as a solution that can produce a meaningful text [8]. However, sometimes RNN experienced exposure bias; it exposed training data distribution instead of its prediction during the training process. Generative Adversarial Network (GAN) was proposed as a generation model using a deep-learning approach. GAN has an adversarial concept between discriminators and generators [9]. GAN has achieved great success in generating realistic synthesis data, including text. However, GAN training also requires lots of data sources to avoid exposure bias or repeated words in sentence results. GPT-2, as a decoder of the Transformer model, is introduced to perform text generation tasks [13].

The Transformer is an auto encoder architecture that uses a self-attention mechanism, making it unnecessary using normal auto encoder cells [11]. Its performance is proven to be fast and accurate compared to standard auto encoders that use RNN. In the self-attention mechanism, GPT-2 works in parallel so that the computation time is used faster than normal RNN. GPT-2 still maintains sequence and meaning, making it able to generate text well even without using cell RNN. GPT-2 can be accessed as both pre-trained and self-trained architecture. Using a pre-trained model that has been trained in large datasets could be useful for our case in the specialized text, Type-3 spam texts as a minor class. However, content and context of Type-3 spam texts are different from training data used for the pre-trained model lead the augmented text does not have a similar context and content.

Fine-tuning is preferable to handle it. Fine-tuning is a method where pre-trained models learn how to better represent the specialized language features by updating weight during backpropagation. In general, fine-tuning step requires more data that finally modifies word representation model. Previous research in representing specialized data involve data addition to work on the word model such as texts for biomedical NLP [14], Autism Spectrum Disorder text-domain [15], and Malware Specific domain [16]. Gu's research evaluates the importance of the dataset size to build word embedding in the Autism Spectrum Disorder case study [15]. The results of the evaluation said that the size of the dataset did not affect. It made the quality of the word embedding from Word2Vec decrease if the dataset used contained many new irrelevant words. This problem can happen because the additional dataset used is abstract research, which often contains new vocabulary and specific terms that are not recognized. Silva proposes a method to expand sentences linguistically on small data to create multiple versions of each sentence to classify sentences with malware specific domains [16]. The method performs better in terms of word similarity when tested on a domain-specific dataset. This method's limitation is that it may not be useful for all words in the representation for contexts defined on a random walk. BioWordVec is proposed to improve the quality of word embedding in biomedical NLP [14]. BioWordVec uses MeSH data (Medical Subject Heading, medical knowledge word dictionary), and the PubMed corpus. MeSH term graphs would be constructed, and sequences of word descriptions would be selected using random sampling. The selected sequence will join PubMed data to create word embedding.

Adding other data such as previous researches above can be provided to be performed in fine-tuning step in text generation. However, not all additional data can be added as facing specialized data such as Type-3 spam text. Those previous method does not selectively add another data concerning context and content aspect. We also have done similar research in content-based filtering to handle mixed-language text models in case of bibliographic information [20]. In this research, we considered the context and content of Type-3 to select additional data. The context of Type-3 is expressed from the structure that the texts

Received: November 11, 2020. Revised: December 11, 2020.

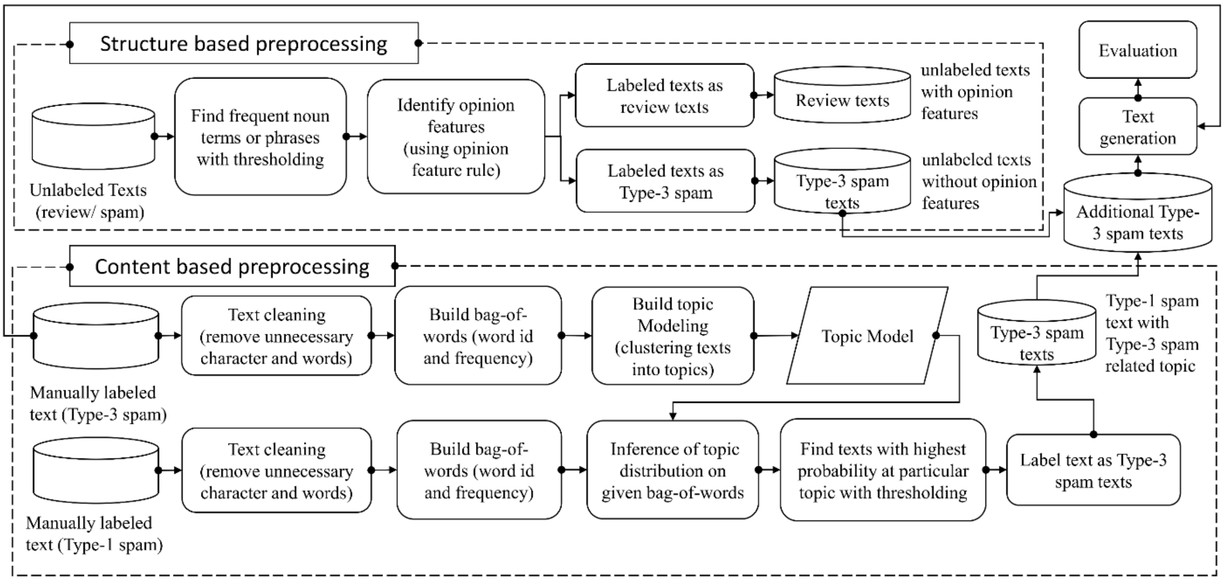
Structure based preprocessing
Unlabeled Texts (review/ spam)
Find frequent noun terms or phrases with thresholding
Identify opinion features (using opinion feature rule)
Labeled texts as review texts
Review texts
unlabeled texts with opinion features
Labeled texts as Type-3 spam
Type-3 spam texts
unlabeled texts without opinion features
Evaluation
Text generation
Additional Type-3 spam texts
Content based preprocessing
Manually labeled text (Type-3 spam)
Text cleaning (remove unnecessary character and words)
Build bag-of-words (word id and frequency)
Build topic Modeling (clustering texts into topics)
Topic Model
Type-3 spam texts
Type-1 spam text with Type-3 spam related topic
Manually labeled text (Type-1 spam)
Text cleaning (remove unnecessary character and words)
Build bag-of-words (word id and frequency)
Inference of topic distribution on given bag-of-words
Find texts with highest probability at particular topic with thresholding
Label text as Type-3 spam texts

Figure. 1 Text generation with structure and content based preprocessing

```
Input : labeled Type-1 Spam text $\{(\boldsymbol{x_i}^{\boldsymbol{1}}, \boldsymbol{y_i}^{\boldsymbol{1}})\}_{\boldsymbol{i=1}}^{\boldsymbol{l}}$,
labeled type -3 text $\{(\boldsymbol{x_j}^{\boldsymbol{3}}, \boldsymbol{y_j}^{\boldsymbol{3}})\}_{\boldsymbol{i=l+1}}^{\boldsymbol{l+u}}$ .
Initially, Let $T^1 = \{(x_i^1, y_i^1)\}_{i=1}^{l}$ and $T^3 = \{(x_j^3, y_j^3)\}_{i=l+1}^{l+u}$
Apply Tokenization and Stop Removal to each $x_i^1$ and $x_j^3$.
Build Bag of Word of $T^1$ and $T^3$
Train Topic Modeling to BOW of $T^3$ and get topics.
Inference Topic Model of $T^3$ to the review $x_i$ in $T^1$
Get topic probability distribution score $S(x)$ of topic $T^3$ in each review $x_i^1$ in $T^1$.
Find highest topic probability score $H(x_i^1)$ in topic probability distribution score $S(x_i^1)$
Set threshold $t$ score topic
If highest topic probability score $H(x_i^1)$ on $x_i^1$ larger than $t$
    Add $x_i^1 \in T^1$ as additional dataset $A^C$
```

Figure. 2 Content-based preprocessing pseudocode

have no opinion features. The proposed method uses an unsupervised approach, topic modeling that can become content revealing of Type-3 spam.

3. Method

In this section, we present the main contribution of this paper. The proposed method is described in Figure. 1. Content-based preprocessing and structure-based preprocessing are two aspects of preprocessing that will be carried out as a data preparation scheme for adding a dataset for text generation deep learning architecture. The content-based preprocessing scheme was inspired by an unsupervised approach using topic modeling to select Type-1 spam reviews based on topic modeling results. The structure-based preprocessing scheme is designed based on the characteristic of Type-3 spam in previous research [3] and uses the opinion feature rule to filtered the unlabeled dataset. Each detailed process of the scheme is described in the next section.

```
Input : unlabeled data $\{(x_i, y_i)\}_{i=1}^{l}$
Initially, Let $U = \{(x_i, y_i)\}_{i=1}^{l}$
Remove exact duplicate data to $x_i$ in $U$
Tag Noun and Noun Phrase to $x_i$ in $U$
Identify Frequent Features of Product
 If product have Frequent Features
  If $x_i$ in $U$ do not mention Frequent Features
    If $x_i$ in $U$ do not have Opinion Rule Tag
      Add $x_i \in U$ as additional dataset $A^s$
 Else
  If $x_i$ in $U$ do not have Opinion Rule Tag
    Add $x_i \in U$ as additional dataset $A^S$
```

Figure. 3 Structure-based preprocessing pseudocode

3.1 Content-based preprocessing

The design of the content-based preprocessing scheme is shown in Figure. 2. This preprocessing concept is selecting other types of spam based on the content of Type-3 expressed by the highest topic modeling distribution result. The highest topic modeling result reveals the adjacency selected reviews to the topic distribution of Type-3 spam. Let $T^1 = \{(x_i^1, y_i^1)\}_{i=1}^{l}$, and $T^3 = \{(x_j^3, y_j^3)\}_{i=l+1}^{l+u}$ define as product review text with Type-1 labeled and product review text with Type-3 labeled, respectively. Type-1 data are labeled based on shingling method and Type-3 spam text are manually labeled. Each x_i^1 review in Type-1 is tokenized and applied stop

Table 1. Opinion feature rule applied in review text

Review Text	Rule	Opinion Feature Extraction
..since dvds have a **wide sound** range, so lower volume sound, like dialog can be difficult ..	JJ, NN	('wide', 'JJ'), ('sound', 'NN')
.. it's nice see apple releasing such a **well-designed successor** ibook at such a reasonable price..		('well-designed', 'JJ'), ('successor', 'NN')
..**battery available** as yet. recharging is easy: i usually recharge my ipod same time i am recharging my notebook battery.	NN, JJ	('battery, 'NN'), ('available', 'JJ')
..basically it was in for repairs at least 5-6 times to replace battery, hard drive, random **reboot issue**, discoloration on plastic, superdrive, etc.		(reboot, 'NN'), (issue, 'JJ')
..although it does collect fingerprints. the **keyboard is awesome**, although spacing feels a little different than my powerbook's does..	NN, VBZ, JJ	('keyboard', 'NN'), ('is', 'VBZ'), ('awesome', 'JJ')
The **display is phenomenal**, The brightness goes very high, much higher than my PowerBook.		('display', 'NN'), ('is', 'VBZ'), ('phenomenal', 'JJ')

removal. Similarly, in each $x_j{}^3$ review of Type-3, it will be formed bag of word.

Bag of word (BOW) is a representation of word occurrences within review text. BOW contains word index information for each word's vocabulary and occurrences. Topic modeling needs input in the BOW form. The number of vocabulary in the bag of words is obtained from the Gensim library's extraction based on the built-in corpus of the library. BOW of T^3 will be trained on topic modeling using Latent Dirichlet Allocation (LDA). Each word will be calculated its word distribution and will be clustered into several topics. LDA will extract the document into several topics based on the BOW of T^3 information**.** The best number of topics is needed to make the best topic model. The calculation of the best number of topics in the LDA method is done by calculating the highest coherence score. The coherence score is obtained from calculating the semantic similarities between high scoring words in a topic.

The result of topic model of T^3 predict topic distribution with probability distribution score $S(x_i{}^1)$ for each $x_i{}^1$ reviews on T^1 spam text. Highest topic probability score $S(x_i{}^1)$ on a topic shows the most probable topic of $x_i{}^1$. $S(x)$ are in the range of 0.0 to 1.0. The closer with a value of 1.0, the most likely the review matches the topic. There needs to be a threshold score of conformity between the review and each topic. The threshold is set arbitrary at 0.7. Each $x_j{}^3$ with a higher $S(x_j)$ than the threshold will be added as additional data A^C of Type-3 spam text T^3.

3.2 Structure-based preprocessing

The design of the structure-based preprocessing scheme is shown in Figure. 3. Real reviews must contain relevant information about the product being reviewed. However, not all information related to the product can be called a review. The information must be the opinion of the product itself or product features. Reviews contain non-opinion of product, such information about shipping, service, shop, and brands conclude to be spam. Structure-based preprocessing is designed to filter unlabeled datasets with no opinion product feature based on opinion rule in part-of-speech structure and took the non-opinion reviews as the other Type-3 spam.

Structure-based preprocessing pays attention to the context of review shown by the structure of sentences in a review. The input of structure base preprocessing is $U = \{(x_i, y_i)\}_{i=1}^{l}$ as unlabeled data. Unlabeled data U were cleaned by removing the exact duplicate of x_i. Any other process was not considered because the whole sentences need to be part-of-speech (POS) tagging. POS tagging is a process of marking the words in a text corresponding to a particular part of speech, based on its definition and context. Structure-based preprocessing has two essential processes, frequent feature identification following by identifying reviews with no opinion rule.

Frequent features of a product are assumed to be the real features of a product because most reviewers will talk about it. Furthermore, Infrequent features are assumed to be the non-related thing about the product. Frequent feature identification uses the first step of the apriori algorithm. Frequent features are obtained from 1-itemsets from a set of reviews in a product that fulfills user-specified minimum support. The number of nouns will be calculated in each x_i product review to form a product's feature set. A noun or noun phrases in reviews tag by part-of-speech tag from NLTK Library can be product features. Brands of the product are removed because it does not include in the product features. A product feature defines as frequent if it appears in more than 1% as

Table 2. Dataset generation used in spam detection

Name	Category	Total
D0	Original Dataset [4]	435
D1	Content-based Preprocessing Result	24
	Structure-based Preprocessing Result	242
D2	Text Generated [13] from Original Dataset	109
D3	**Text Generated from Proposed Method**	**109**

minimum support in a product review.

Opinion features are identified with the predefined rule of part-of-speech called opinion rule. There are three opinion rules to get a feature opinion phrase. Table 1 show some examples of reviews that have opinion phrases. A product feature is usually a noun or noun phrase. Reviewers express opinions on the product's features using adjectives ‘JJ’ on objects ‘NN’. Usually, the ‘NN’ tag is preceded by the ‘JJ’ tag or vice versa. Nouns can also be described directly, such as, for example *(‘keyboard is awesome’*). The noun ‘NN’ is followed by ‘VBZ’ (as a description) and followed by an adjective ‘JJ’ that describes the object's nature. In some cases, there will be an implicit sentence like (“*I like it”)*. However, this research will only focus on reviews that have explicit intentions. The opinion rule described above will be used as a filter to retrieve non-opinion reviews text. The noun and noun phrases were taken to be the set of features.

The product review texts are then divided into two parts. The first is products that have frequents features in the review. The second is products that do not have frequent features. From products having frequent features, some reviews do not mention or comment on the frequent features. Those reviews that do not mention frequent features and contain no opinion features are marked as additional data of Type-3 spam A^S. All reviews contain no opinion features marked as additional data of Type-3 spam text A^S from products that do not have frequent features.

3.3 Spam text generation with pre-trained model

The text generation architecture uses some pre-trained Transformer models. Each layer in Transformer model uses a self-attention mechanism and a mask head to maintain the sequence in the review. Instantiating a pre-train model will create a model instance with weights copied from the pre-trained model. In addition the experiment also conducted in RNNs model for comparison, Type-3 spam T^3 and additional data A^S, A^C are only used as training data in RNNs model. The pre-trained model trained on Type-3 spam T^3 and additional data A^S, A^C will be processed in fine-tuning steps. While fine-tuning the model, the pre-trained network's weights will be updated during backpropagation based on the information in fine-tune data.

4. Result and analysis

4.1 Data preparation

This research uses the Amazon Product Review Data dataset. The dataset is the result of crawling the Amazon web. The product being reviewed is 21 products of manufactured products such as electronic goods, laptops, cameras, cd players. Total all of the product reviews are 42336. Those data were not labeled. Labeling Type-1 data were done by discovering duplicate data since Type-1 data can not detect by manually. Duplicate data are obtained by calculating the Jaccard similarity of reviews to one another with a threshold. The threshold is determined arbitrarily at 0.9. One hundred twenty-six reviews, which are near-duplicate reviews, were labeled as Type-1 spam. Type-3 spam was obtained by manual labeling producing 100 data. The data labeled with Type-2 spam that have been manually labeled are very few, just ten reviews. They are really rare to be found manually. Hence, Type-2 was assumed as Type-3. In the next step, types-3 and Type-1 was used in content and structure-based preprocessing to produce additional Type-3 spam text for text generation. Type-3 and additional Type-3 spam text will be trained for the text generation model.

4.2 Experiment setup

There are two experiments conducted in this research, text generation and spam detection. The purpose of the experiment on text generation is to discover the impact proposed method on generated text compared to the existing method. The existing text generation model tested are RNNs [8] and pre-trained Transformer model [12, 13] with and without proposed content and structure-based preprocessing. Evaluation measures by selecting a seed sequence in original Type-3 spam text and generated text, then calculate their correspondence using BiLingual Evaluation Understudy Score (BLEU).

BLEU is a metric introduced to measure quality of machine translation represented by the correspondence between translated (candidate) sentences and human-translated (reference)

Table 3. Top 10 keywords of topic modeling result

ID	Top 10 Keywords	Type-1 spam categorized in Type-3 spam's topic	Topic Description
1	using, driver, identify, match, server, course, unit, create, application, hours.	16 Type-1 spam text categorized as Topic 5.	Most words are used to promote a CD of training.
2	identify, servlet, course, code, using, specific, situation, java, used, application,	None of Type-1 spam text categorized as Topic 2.	
3	jdbc, using, identify, database, create, data, course, features, code, hours.	1 Type-1 spam text categorized as Topic 3	
4	ipod, player, great, get, one, using, like, buy, use, bought.	7 Type-1 spam text categorized as Topic 4	Topic contains opinion words to brand, such as Apple, Amazon services and shipping.
5	love, product, one, would, got, never, amazon, item, identify, order.	None of Type-1 spam text categorized as Topic 5.	

Table 4. Unlabeled data partition in structure-based preprocessing

Product Category	Total	Opinion Review	Non Opinion Review	Description Data
Have Frequent Item	1360	1304	56	- From 4701 reviews of product having frequent item, only 1360 reviews that do not mention the frequent item in review text and 3341 reviews mention the frequent features. - From 1360 review, only 56 reviews that do not contain opinion rule in the text structure of POS Tagging.
	3341	3307	34	- From 3341 reviews mentioning the frequent features, 34 reviews do not contain opinion rule in the text structure of POS Tagging.
Don't Have Frequent Item	9726	9564	162	- From 9726 reviews of the product that do not have frequent items, only 162 reviews that do not contain opinion rule in the review text.
Total	14427	14175	252	- From 14427 total reviews, only 162 reviews that do not contain opinion rule in the review text

Table 5. Evaluation on text generation result of proposed method comparing with other latest method

Preprocessing	Text Generation Architecture	BLEU Score		Eval Loss
		Mean	Standard Deviation	
Without Proposed Content and Structure-based Preprocessing	RNN [8]	0.122	0.175	17.55
	LSTM	0.145	0.185	14.86
	GRU	0.021	0.056	16.56
	OpenAI-GPT [12]	0.363	0.140	4.46
	GPT-2 [13]	0.347	0.110	3.66
	GPT-2 Medium [13]	0.366	0.139	3.57
With Proposed Content and Structure-based Preprocessing	RNN	0.147	0.180	14.88
	LSTM	0.177	0.208	17.41
	GRU	0.160	0.195	18.89
	OpenAI-GPT	0.365	0.142	4.29
	GPT-2	**0.373**	**0.112**	**3.98**
	GPT-2 Medium	0.370	0.110	3.67

sentences. BLEU was originally used for case studies of translation done by machine, but BLEU can also be applied to other language model problems such as text generation, image caption generation, text summaries, and speech recognition.

The BLEU score formula is shown in Eq. (1). The BLEU score is obtained by calculating the geometric mean of the modified corpus precision value p_n of the sequence test, then multiplying it by the brevity penalty factor BP. Brevity penalty BP is obtained

Table 6. Sample text generation result from proposed method comparing with original text and existing method

Original Type-3 spam text	1	Why is everyone rating this product a "1" just because Amazon screwed up your order?? Your rating has nothing to do with the product itself **and makes the product look bad. Maybe I should rate this product a "1" because Amazon ran out of my body wash…**
	2	I placed my order on 26 Sept. 2005. Just now received notification of another delay with shipping sometime between **4 and 15 Nov 2005. Do the math........that's approximately 7 weeks for those that can't count!**
	3	I would have liked to review this item, but, unfortunately, it got stolen before **I could even see it. It was stolen out of the box, from my front porch.**
Text Generation Result from Proposed Method	1	Why is everyone rating this product a "1" just because Amazon screwed up your order?? Your rating has nothing to do with the product **and everything will be fine for now, but that's another post..i got my dog back soon after getting it i think at time im going...**
	2	I placed my order on 26 Sept. 2005. Just now received notification of another delay with shipping sometime **before 27 Nov 2006 (due to bad weather).**
	3	I would have liked to review this item, but, unfortunately, it got stolen before **I could even try and use the product.**
Text Generation Result from Existing Method [13]	1	Why is everyone rating this product a "1" just because Amazon screwed up your order?? Your rating has nothing to do with the product itself! **- Jan 3, 2013This thing works great in my 1999 Suzuki.**
	2	I placed my order on 26 Sept. 2005. Just now received notification of another delay with shipping sometime **within 24 hours...**
	3	I would have liked to review this item, but, unfortunately, it got stolen before **I could place my order**

from the condition of number of words from the generated text result c and the number of original text r described in Eq. (2). The p_n modified precision score is obtained from the number of n-grams of the text generation result similar to the original text divided by the number of n-grams of generation result. Meanwhile, the weight w_n obtained from the sum of $1/N$ (the standard N value for BLEU is 4). The best value for BLEU is 1.0, and the lowest is 0.0. The closer BLEU score with 1 indicates the more similar generated text with the original text. A score of 1 is infrequent because of generated text from machine commonly are not the same original human writing text.

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \quad (1)$$

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-\frac{r}{c})} & \text{if } c \leq r \end{cases} \quad (2)$$

Other experiment, spam detection is also executed by comparing data generation scenario in four classification models. This experiment is carried out to see the effectiveness of generated text from the proposed method on several spam detection models. The scenario of the combination of data from the generated text used is detailed in Table 2. D0 is the original imbalanced data from manually labeled contains 109 Type-3 spam and 326 non-spam [4]. D1 is data from content-based and structure-based as additional data also tested to see whether the data can significantly impact spam detection. D2 is the result of the existing text generation model [13] without proposed content and structure-based preprocessing. The existing model used is the one that has the highest BLEU score among others in the previous experiment. D3 is the result of our text generation method with content and structure-based preprocessing.

Feature extraction used is TF-IDF in unigram and bigram text. Parameters used in the classification model are Gaussian Naïve Bayes for Naïve Bayes, kernel linear for Support Vector Machine, k=5 for KNN, which has the best result, among other k. MLP Classifier uses two hidden layers, activation function ReLU, quasi-Newton optimizer, which can converge faster and perform better for small data. The evaluation in spam detection measures F1-score and recall of Type-3 spam class with cross-validation. The best value for F1-score and recall is 1.0, and the lowest is 0.0. F1-score carries the harmonic mean signifying the balance between the precision and the recall of Type-3 spam. Precision is the fraction of the actual Type-3 spam instances among the detected instances, while recall is the fraction of the total amount of actual Type-3 spam instances detected [21]. The higher the recall score, the more actual Type-3 spam texts were successfully detected by spam detection. The higher F1-score signifies the higher both precision and recall.

4.3 Content and structure-based preprocessing results

Content-based preprocessing performs topic modeling of Type-3 spam at the first stage producing five topics. Five top topics with ten top keywords and topic inferencing of Type-1 spam in Type-3 spam's topic result are shown in Table 3. It is reasonable that most topics reveal words that use in an advertisement. Because 20% of Type-3 spam text contain advertisement review, and they are long texts. Those reviews have similar usage of words in more than one product.

Although non-related things are not exposed to the top keyword in every topic, topic five can represent non-related things about the product. The result of topic inferences turns out to have total 24 of Type-1 spam text that fulfills the threshold of topic probability score. Hence, 24 additional Type-3 spam texts are produced from content-based preprocessing. Those data were categorized with the same content as Type-3 spam based on topic inferencing and become additional data of Type-3 spam.

Removing duplicate in the unlabeled dataset is the first step in structure-based preprocessing. After removing duplicates, 42336 reviews are cleaned, resulting and build partition that has been processed in structure-based preprocessing shown in Table 4. In data that have mention frequent item, there may be a review that explicitly says opinion about the feature *"I love the monitor"*. Even though those reviews do not mention the monitor's specialized opinion, it is acceptable that those are non-spam. Those phrases do not filter by the opinion rule explained in Table 1. However, there may be a text, "*I receive the camera as a gift*". The text mentioned the camera as a frequent feature but did not mention the comment of the features. In this phase, manually traced further are suggested. Reviews that do not mention frequent item and have no opinion rule were categorized as additional data from structure-based preprocessing since infrequent features were assumed as non-related things about products.

From the structure-based preprocessing, there are 252 additional Type-3 spam and 24 additional data of Type-3 spam from content-based preprocessing. The reviews with words' count less than ten words are removed to make the text generation model understand clearly through review sequences. Final data are 266 after removing text less than ten words.

4.4 Text generation results

Type-3 spam and the additional data from preprocessing are used for data training to be processed in training from scratch or fine tuning steps in text generation according to the architecture used. RNN, LSTM, and GRU architecture which not have a pre-trained model use training data to train the model from scratch. For the pre-trained model such as OpenAI-GPT and GPT-2, the training data perform fine-tuning steps. The total data 375 from 109 Type-3 spam and 266 from additional data is divided into training data and test data, and validation data by splitting 80%, 10%, and 10%. The text generation model produces 109 data by generating text from the seed of original Type-3 spam. Table 5 shows the average BLEU score in 109 Type-3 spam after text generation in various text generation model with and without proposed preprocessing.

The latest method, the RNNs generation model [8] achieves the lowest BLEU score and highest validation loss. The small data as training data from scratch cause the model to produce low-quality text that is not similar to Type-3 spam. Text generation results from the Transformer model [12, 13] performs better than RNNs because they already have a model trained in large data before. Text generation with proposed content and structure-based preprocessing perform better with a higher 0.038 scores on average BLEU scores than six existing text generation models without proposed preprocessing. This low score improvement could be caused by the Transformer model have been pre-trained with data from a large document with a total text of 40 GB. So adding 266 data has a low impact. Though not significant, it shows that the proposed method produces more similar text than the existing method.

Table 6 shows the sample text generation result from proposed method comparing with existing method and original text. The bold font in texts show the original text and generated text produces from method. On the text number 1 and 3, proposed method generate more reasonable text related to the previous text. The proposed method also generates texts longer than the existing method and maintains each review's content. The generated text also does not have an opinion rules structure. The existing method contrary produces unrelated text that fails to represent generated text that similar to original Type-3 spam.

4.5 Spam detection results

Table 7 shows the spam detection results on various datasets shown in Table 2 that have been

Table 7. Spam detection results with additional data from our text generation method to illustrate the effectiveness the proposed preprocessing steps

Dataset Generation	Classicication Model	F1-score (%)	Recall (%)
D0 [4]	NB	15.04	0.09
D0, D1		16.11	14.4
D0, D1, D2		46.12	50.72
D0, D1, D3		47.68	53.4
D0, D2 [13]		69.1	93.11
D0, D3		**71.49**	**98.18**
D0 [4]	SVM	85.38	81.36
D0, D1		92.09	93.84
D0, D1, D2		94.32	96.88
D0, D1, D3		95.04	97.3
D0, D2 [13]		95.48	97.22
D0, D3		**96.01**	**98.63**
D0 [4]	KNN	21.66	16.36
D0, D1		28.76	17.83
D0, D1, D2		75.90	69.39
D0, D1, D3		48.22	34.25
D0, D2 [13]		**86.88**	**86.94**
D0, D3		48.16	36.14
D0 [4]	MLP	73.81	65.00
D0, D1		73.43	72.99
D0, D1, D2		93.94	94.17
D0, D1, D3		**94.72**	**95.22**
D0, D2 [13]		93.68	93.11
D0, D3		90.64	94.94

explained in Experiment Setup, which include additional data from our text generation method to illustrate the effect of the proposed content and structure preprocessing steps. The detection result on Naïve Bayes, KNN, and Multi-Layer Perceptron (MLP) of D0 has a low F1-score and spam recall. It can be caused by the imbalanced data that has very few spams labeled.

The detection results using additional data D3 gave the highest F1-score and recall in Naïve Bayes and SVM classification methods. Meanwhile, in MLP, a combination of D0, D1, and D3 gives the highest F1-score and recall. D3 does not give better results in KNN. The addition of D2 significantly has an impact on F1-score and recall. D2 results from the existing text generation method produce shorter text shown in Table 6 than the proposed method D3. Those shorter texts are assumable as duplication of D0 data because the more same texts are used in D2. KNN model can be sensitive to the high variation of data D3 that contains longer and more new words and work better in similar words of text used in D2. Thus, KNN gives better results in the addition of D2 than D3.

4.6 Discussion

The experiment in various text generation models exposes the need for the proposed content and structure-based preprocessing in text generation producing text more identical Type-3 spam marked by higher BLEU score and actual generated text. Augmented data from proposed method help spam detection to retrieve Type-3 spam better than spam detection with imbalanced data marked by improved recall score and F1-score. The effectiveness of this work is justified by the effect of the proposed method on generated text and the effect of generated text on spam detection result. From those two experiments, the proposed method's effectiveness is producing text more related to original Type-3 spam text and impacts significantly to build better spam detection.

The result of this research demonstrate that the proposed method improves the metric value of 0.038, called BLEU (Bi-Lingual Evaluation Understudy). Though the improvement is not significant, the sample generated text from the proposed method has more correspondence text maintaining the context and content of Type-3 spam texts. From spam detection results, (D0, D3) and (D0, D1, D3) are the combination data generation that gives the highest recall and F1-score in spam detection results. Combination of original data D0 and augmented data D3 from the proposed method have 41.27 % higher recall score and 27.6 % higher F1-Score on average than original dataset D0 that imbalanced.

5. Conclusion

Our contribution in this paper is proposing a preprocessing mechanism to maintain the content and structure of the generated text in an imbalanced dataset. The proposed method explores opinion feature structure from the unlabeled dataset and inference data from Type-1 spam review based on the topic of Type-3 spam. Based on our closing in discussion it is stated that the result of our work receive higher BLEU score indicating the result of generated text from proposed method producing text more correspondence with original text. Another result of our work is generating text from the proposed method as augmentation data also improve spam detection result. Future work of this research can be done by concerning other content revealing steps such as applying semantical representation. Also, considering other opinion rules or steps to cover implicit opinion sentences.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Ana Alimatus Zaqiyah: conceptualization, methodology, formal analysis, Investigation, writing—original draft preparation and editing. Diana Purwitasari: supervision, validation, formal analysis, writing—review and editing, funding acquisition. Chastine Fatichah: supervision, formal analysis, writing—review.

Acknowledgments

This research was supported by a research funding scheme for Informatics Engineering Department, Institut Teknologi Sepuluh Nopember with No. 1666/PKS/ITS/2020.

References

[1] G. Stanton and A. A. Irissappane, “GANs for Semi-supervised Opinion Spam Detection”, In: *Proc. of International Joint Conf. on Artificial Intelligence (IJCAI)*, pp. 5204–5210, Macao, China, 2019.

[2] N. Archak, A. Ghose, and P. G. Ipeirotis, “Deriving the Pricing Power of Product Features by Mining Consumer Reviews”, *Management Science*, Vol. 57, No. 8, pp. 1485–1509, 2011.

[3] B. Liu, *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Berlin, Heidelberg: Springer Heidelberg, pp. 459–526, 2011.

[4] D. A. Navastara, A. A. Zaqiyah, and C. Fatichah, “Opinion Spam Detection in Product Reviews Using Self-Training Semi-Supervised Learning Approach”, In: *Proc. of 2019 International Conf. on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)*, Batu, Indonesia, pp. 169–173, 2019.

[5] J. Li, M. Ott, C. Cardie, and E. Hovy, “Towards a General Rule for Identifying Deceptive Opinion Spam”, In: *Proc. of the 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, USA, pp.1566–1576, 2014.

[6] D. Hernández Fusilier, M. Montes-y-Gómez, P. Rosso, and R. Guzmán Cabrera, “Detecting positive and negative deceptive opinions using PU-learning”, *Information Processing & Management*, Vol. 51, No. 4, pp. 433–443, 2015.

[7] A. Ghazal, T. Rabl, M. Hu, F. Raab, M. Poess, A. Crolotte, and H.-A. Jacobsen, “BigBench: Towards an Industry Standard Benchmark for Big Data Analytics”, In: *Proc. of the 2013 Association for Computing Machinery (ACM) Special Interest Group on Management of Data (SIGMOD) International Conf. on Management of Data*, New York City, USA, pp. 1197–1208, 2013.

[8] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation”, In: *Proc. of the 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, pp. 1724–1734, 2014.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative Adversarial Nets”, In: *Proc. of Advances in Neural Information Processing Systems*, Montreal, Canada, pp. 2672–2680, 2014.

[10] J. Chen, Y. Wu, C. Jia, H. Zheng, and G. Huang, “Customizable Text Generation via Conditional Text Generative Adversarial Network”, *Neurocomputing*, Vol. 416, pp. 125–135, 2020.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is All You Need”, In: *Proc. of Advances in Neural Information Processing Systems*, Long Beach, USA, pp. 5998–6008, 2017.

[12] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving Language Understanding by Generative Pre-Training”, *Technical Report, OpenAI*, pp. 1–12, 2018.

[13] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners”, *OpenAI Blog*, Vol. 1, No. 8, pp. 9–33, 2019.

[14] Y. Zhang, Q. Chen, Z. Yang, H. Lin, and Z. Lu, “BioWordVec, Improving Biomedical Word Embeddings with Subword Information and MeSH”, *Scientific Data*, Vol. 6, No. 1, pp. 1–9, 2019.

[15] Y. Gu, G. Leroy, S. Pettygrove, M. K. Galindo, and M. Kurzius-Spencer, "Optimizing Corpus Creation for Training Word Embedding in Low Resource Domains: A Case Study in Autism Spectrum Disorder (ASD)", In: *Proc. of American Medical Informatics Association (AMIA) Annual Symposium*, San Francisco, USA, pp. 508–517, 2018.

[16] A. Silva and C. Amarathunga, "On Learning Word Embeddings From Linguistically Augmented Text Corpora", In: *Proc. of the 13th International Conf. on Computational Semantics*, Gothenburg, Sweden, pp. 52–58, 2019.

[17] T. Fahrudin, J. Buliali, and C. Fatichah, "Enhancing the Performance of SMOTE Agorithm by Using Attribute Weighting Scheme and New Selective Sampling Method for Imbalanced Data Set", *International Journal of Innovative Computing, Information and Control*, Vol. 15, pp. 423–444, 2019.

[18] A. Setiyoutami, W. Anggraeni, D. Purwitasari, E. M. Yuniarno, and M. H. Purnomo, "Extracting Temporal-Based Spatial Features in Imbalanced Data for Predicting Dengue Virus Transmission", In: *Proc. of Advances in Computer, Communication and Computational Sciences*, Bangkok, Thailand, pp. 731–742, 2021.

[19] S. Akkaradamrongrat, P. Kachamas, and S. Sinthupinyo, "Text Generation for Imbalanced Text Classification", In: *Proc. of 2019 16th International. Joint Conf. on Computer Science and Software Engineering (JCSSE)*, Pattaya, Thailand, pp. 181–186, 2019.

[20] D. Purwitasari, C. Fatichah, I. K. E. Purnama, S. Sumpeno, and M. H. Purnomo, "Inter-departmental Research Collaboration Recommender System Based on Content Filtering in A Cold Start Problem", In: *Proc. of 2017 IEEE 10th International Workshop on Computational Intelligence and Applications (IWCIA)*, Hiroshima, Japan, pp. 177–184, 2017.

[21] P. Flach and M. Kull, "Precision-Recall-Gain Curves: PR Analysis Done Right", In: *Proc. of Advances in Neural Information Processing Systems*, Montreal, Canada, Vol. 28, pp. 838–846, 2015.