Automated Text Augmentation with Web-Based Interface Application Using Flask Framework for Indonesian Text

I A Rahma[1], **L H Suadaa**[2]

[1] BPS-Statistics Indonesia, Tabalong Regency, Indonesia
[2] STIS Polytechnic of Statistics, Indonesia

*Corresponding author's e-mail: lya@stis.ac.id

Abstract. In real world, data and resources available for text classification are limited. One of issues on labelled data is imbalanced data. Problem of imbalanced data affects performance and accuracy of model because the model only focuses on data with majority label. Therefore, the measure of model accuracy cannot describe the true quality of model. To overcome this, an oversampling approach is carried out. Text-based oversampling is known as text augmentation. However, NLP resources for Indonesian, especially in performing text augmentation, are still limited. Therefore, this research conducts development of a web application to augment Indonesian text automatically. The application was bulit using prototype method. The application was successfully built and can facilitate users to perform augmentation automatically for all texts in the dataset. Users can select preferred augmentation technique and are required to upload datasets as input. The output of application is same dataset file as input with an additional column containing synthetic text augmented by the application. This application can contribute to further research in performing text augmentation for Indonesians.

1. Introduction

Modernization that occurs in this digital era brings changes to human activities. Digitized human activities produce digital traces and data [1]. The generated data has a large volume, increases in a short time, high diversity, and advanced processing to get value and information in it. Data with these characteristics is referred to as big data. The potential of big data which can be a source of information makes experts and researchers intensively continue to improve the quality of big data. One type of data in big data is text data. Text data analysis has an important role in various fields in government and business. Processing and analysis of text data is carried out using natural language processing (NLP), which is on of branch of artificial intelligence (AI). NLP can process text automatically like humans do [2].

One of the fundamental tasks in NLP is text classification. Text classification is commonly applied to filter spam emails, sentiment analysis, and hate speech detection [3]. Classification is supervised learning and thus requires a dataset for the training process. However, the data obtained from actual cases in the real world are often imperfect to support the training process so that it requires various additional processing before being elaborated. One of the issues related to datasets for text classification is imbalanced data. In imbalanced data condition, the number of instances between labels is significantly different [4][5][6]. Imbalanced data needs to be a concern and resolved before classifying. This is

because in imbalanced data, the distribution of sample data for training becomes imbalanced so that classification will tend to ignore labels with a small number and focus on labels with large instances. The model's ability to predict correctly is valid for majority label only. In fact, for some cases, the ability of the model to correctly predict minority labels is more important, for example in spam, fraud, and hate speech detection. Where instances with these labels generally have a smaller number in real world. Even high accuracy values cannot be used as an illustration of the model's performance because the model mostly correctly classifies only the majority labels [6][7].

Handling imbalanced data has two approaches, which are internal and external approaches [8]. The external approach is considered more adaptive compared to the internal approach because the internal approach is done by modifying the algorithm to adjust the distribution and characteristics of dataset so that it causes problems if applied to different datasets. Meanwhile, external approach uses the concept of resampling to deal with dataset imbalance. Possible solutions are undersampling and oversampling. The oversampling approach means increasing the samples of the minority labels to make it close to the size of the majority labels, while the undersampling approach means reducing the samples of majority labels to make it close to the size of minority labels [8]. In case of text classification, oversampling with SMOTE method is widely used to handle imbalanced data. This is proven by research that obtained result that SMOTE is better than other undersampling or oversampling methods [9]. However, application of SMOTE on text data cannot be done on the original text, but on text features [10] or their representations [11]. Therefore, the additional text data from SMOTE results are not good in language and context from the original data labels so that they do not overcome textual data [12]. SMOTE does not pay attention to the semantics and syntax of the original text examples [13]. In fact, semantics and syntax are very important in text processing to obtain context and meaning.

To overcome the shortcomings of SMOTE in handling unbalanced text data, another alternative is needed. As mentioned before, context is very important in text processing. The role of NLP becomes very important here. However, there are limitations in NLP resources and systems to do so, especially in Indonesian. Indonesian is one of the low-resource languages in the field of NLP [14]. In fact, according to Internet World Stats, Indonesian is the fourth most used language on the internet with a total of around 171 million users worldwide. However, the progress of NLP research in Indonesia has been slow [15]. It creates a problem because the availability of good datasets plays an important role in determining model performance and accuracy. Dataset development is necessary, but it also requires adequate resources.

One alternative to improve the quality of text datasets is text augmentation. Text augmentation can be used as one of the alternative techniques to support the development of NLP resources [16]. The main challenge of applying text augmentation to text classification is how to generate new text without affecting the original label [17]. English and Chinese text augmentation research was conducted with the help of Thesaurus to select synonyms as substitute words [18]. The application of synonym replacement was also carried out to improve classification performance on Indonesian datasets [19]. The method used is synonym-based text augmentation, which replaces one or several words in a sentence with their synonyms. Replacement of synonyms to produce new text data for English is done by adding a POS tagging (part of speech tagging) task to find out the class of words so that the synonyms obtained are the right equivalent and in accordance with the context of the sentence [19][20]. Another more advanced and practical augmentation technique for English utilizes Cloud API NLP [21]. In this study, SyntaxNet, Google Translate API, and various augmentation techniques were used.

The development of text augmentation for English is so rapid that there are many resources to make text augmentation easier. For Indonesians, the application of text augmentation has not been done much. This is related to the availability of NLP resources. Therefore, research in the field of Indonesians NLP, especially text augmentation, has potential to continue to be developed. In line with this, many text classification cases face the problem of imbalanced data. In fact, as mentioned earlier, imbalanced text data needs to be handled before further processing so that the model and classification results are truly accurate and reliable. Handling imbalanced data becomes important when these problems arise in the research conducted. However, researchers need to learn and explore more about handling imbalanced

data up to text augmentation. The research becomes broader because it should not be included in the domain and focus of the research. The process can take more time and effort. In addition, text augmentation tools for Indonesian language are still limited and inadequate. The lack of resources for text augmentation in Indonesian is the motivation for this research. The existence of an automatic text augmentation application for Indonesian can be one of the text augmentation tools in helping and facilitating researchers when facing unbalanced data problems. With automation in performing text augmentation, the time in processing text data becomes faster, more practical, and easier, researchers do not have to think about the complicated concepts and techniques that run behind it.

Based on the description that has been explained, this research focuses on the problem of how text augmentation for Indonesian can be done easily, practically, and saves time. Therefore, the purpose of this research is to design and build a web application for text augmentation in Indonesian automatically. The results of the research are expected to contribute to future research by helping to perform the text augmentation process more simply and without having to think about the complicated process behind it.

2. Methods

2.1. Research scope

This research develops the field of NLP in performing text augmentation for Indonesian language by designing and building a web-based application for automated text augmentation. The augmentation technique used is based on synonym replacement and back translation. There are three techniques applied which are explained below. All techniques were initiated by the author in his previous research [22].

1. *Synonym replacement*

The synonym replacement technique [20] uses POS tagging and Thesaurus. POS tagging is performed using the CRF model [23] to detect class of each word in a sentence. Meanwhile, the thesaurus used is Indonesian Thematic Thesaurus by Language Development and Guidance Agency of the Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia as a synonym dictionary. The thesaurus can be accessed via web with the link https://tesaurus.kemdikbud.go.id/tematis/ using Python module. Synonym replacement is done based on the combination of word tag with word. If the combination has a synonym in Thesaurus, then word can be replaced with its synonym. Otherwise, the original word will still be used. First label in Thesaurus will be selected on the grounds that it is most widely used. If more than one synonym is found, then randomisation is performed.

2. *Synonym replacement with optimization*

The synonym replacement with optimization technique has the same working principle as the synonym replacement technique. The optimization is done by using Fasttext Indonesian word vector model [16] to measure similarity between original word and its synonyms. All synonyms obtained from Thesaurus will be checked for their similarities with the original word. The selected synonyms must have a similarity of 0.5 [23][24]. If there is more than one synonym, then randomisation is performed. This technique aims to improve the previous synonym replacement technique in selecting synonyms. The selected synonyms really have semantic closeness to the original word.

3. *Back translation*

The augmentation technique with back translation utilizes freely accessible Google Translate API to translate text. Back translation has the concept of paraphrasing by translating the original text into another language which is referred to as an intermediate language and then translate it back into Indonesian. There are five intermediate languages used in this augmentation application, namely English, Chinese, Malay, Javanese, and Tagalog. The selection of languages is based on the intensity of the use of languages other than Indonesian by the Indonesian people and the proximity of languages and the findings of previous research [24].

Web applications built using text augmentation techniques are suitable for short texts. Long text augmentation is still possible. However, the results are not good, especially when using techniques with synonym replacement bases that do not pay attention to punctuation. The alternative that can be chosen is to use the back translation technique when augmenting long texts.

2.2. Application design and development

The purpose of developing this augmentation application is to be able to perform augmentation automatically without needing to pay attention to complicated process behind it. For this reason, the application built is a web-based interface application that performs augmentation based on several augmentation techniques provided. The general design of the augmentation application can be seen in Figure 1. To be able to perform automatic augmentation on web application, several inputs are required, that are augmentation technique, intermediate language, and dataset. Then, the web application will run augmentation process. The output given to the user is same dataset file as input with an additional column containing augmented text.

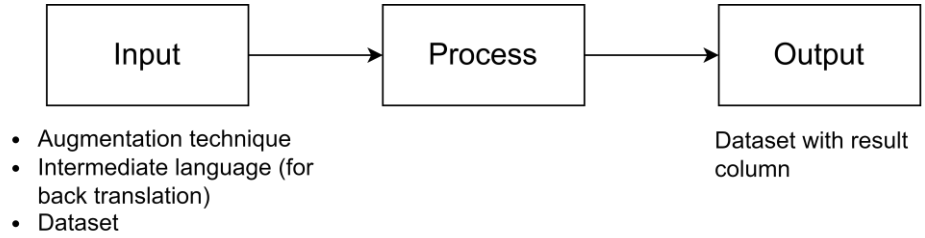

Figure 1. Input output design.

The web application was built using Flask as a web development framework accessed with Python and HTML, CSS, and Javascript programming languages for web development and applied using Jupyter Notebook. Application development is carried out using the prototype method. Prototyping is a system development process that uses a prototype approach [25]. Prototype is defined as a tool that gives developers and users an idea of how the system functions in its complete form. The development and testing of prototypes of applications made is done quickly through interactions and iterative processes commonly used by information systems experts and business experts. The prototype method allows users to have a basic overview of the program and do initial testing. Prototyping has the concept of simplifying and accelerating system design. This method is best used when the requirements definition is not yet clear in detail [25].

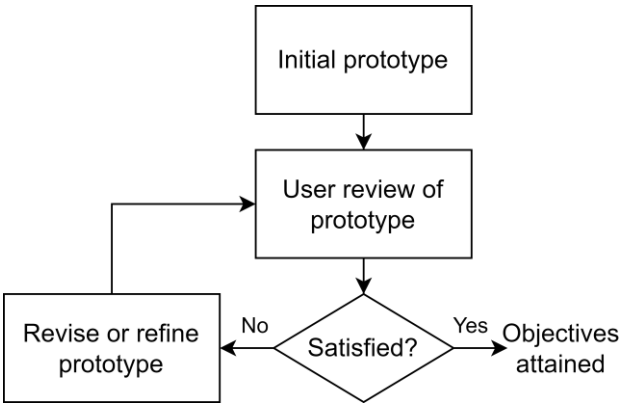

Figure 2. Prototyping process [26].

2.3. Implementation and testing

The web application that has been built according to the design is implemented using the original dataset. The datasets used have two different characteristics. The first dataset contains text with formal language. The formal text dataset used is Indonesian online news headline text (CLICK-ID Dataset) [27]. Meanwhile, the second dataset contains texts with informal language which is colloquial language containing unstandardized words or slang words such as unstandardized writing, abbreviations, typos, and others. The informal text dataset used is a user comment text on an Indonesian online news portal [9].

3. Results

3.1. Application Design

The web-based interface application built has the function of performing the dataset augmentation process automatically. The application input is in the form of .xls or .xlsx format files consisting of at least two columns, namely id and text. The writing of the column names must be the same and will be validated in the application which will be explained next. Users can choose preferred augmentation technique among three techniques, that are synonym replacement, synonym replacement with optimization, and back translation. Output of the application is the same file as input file with addition of gen_text column which is the augmented synthetic text.

The augmentation application created has one main feature which is generating text. Figure 3 shows the use case diagram of the application. Text generation feature has functions tied to it. When generating text, the function of selecting a technique, uploading a dataset, and going to the download page must be executed. Then, there is one function that is an extension of the action of selecting a technique, when selecting a technique then there is a possibility of the select language action being performed. Select language is performed when the user chooses the back translation technique.

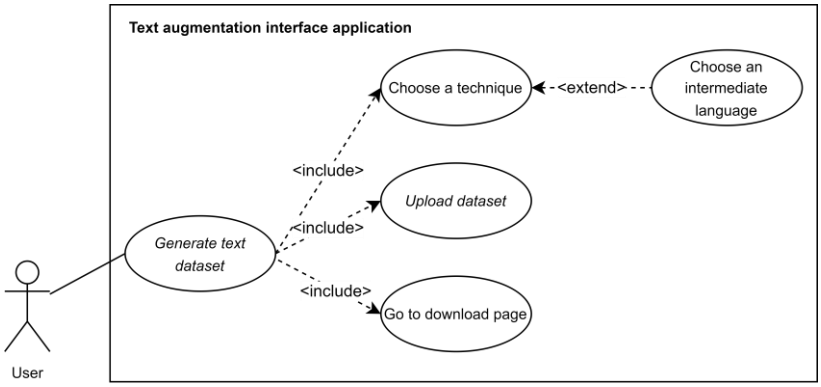

Figure 3. Use case diagram for augmentation application.

The process flow when the application is running based on the use cases that have been created previously can be seen through the activity diagram in Figure 4. When the user enters the main page, instructions will be displayed to select an augmentation technique and upload a dataset. If the user chooses back translation augmentation technique, language options will be displayed for the translation process. The user is instructed to select one language. Then, the user can generate text if they have uploaded the dataset. If dataset has not been uploaded correctly, the validation process will display a command to upload the dataset first or correct the data structure in dataset. After that, the augmentation process can be run. After a while, the augmentation is complete. The user will be redirected to the download page. On the download page, the user can download the augmented dataset and perform augmentation again. If the user chooses to perform augmentation again, the user will be taken to the main page and can augment again in the same way.

3.2. Application Interface and Overview

On the main page that can be seen in Figure 5, the user is given a visualization to enter input in the form of a selection of available augmentation techniques and a box for uploading the dataset to be augmented. In the technique selection, information about the estimated processing time calculated for short text is also displayed. In addition, the box for uploading the dataset provides information about the specified dataset file format, namely .xls or .xlsx. The dataset must have at least a single text and id column, where text is the text to be augmented and id is the index of each text. The name and writing of the column must be as specified. When input has been entered, the user simply presses the generate text button to perform augmentation process. The augmentation process runs only for one augmentation. This considers the time for one augmentation which is quite long if the dataset is large. Furthermore, when the user chooses the back translation augmentation technique, the application displays information about the selected language and language options as an intermediate language that can be selected. For one augmentation process, the user is asked to choose only one language. The display when selecting the back translation technique can be seen in Figure 6. Then, when the user switches from back translation to another technique, the language selection will disappear and the selected language information is still displayed. This is done to help users and not confuse them. The input of augmentation techniques and language options when choosing back translation applies a default value so that even if the user does not choose, the augmentation process can continue as long as the dataset file has been uploaded. The default value for augmentation technique is synonym replacement and for language option is English.

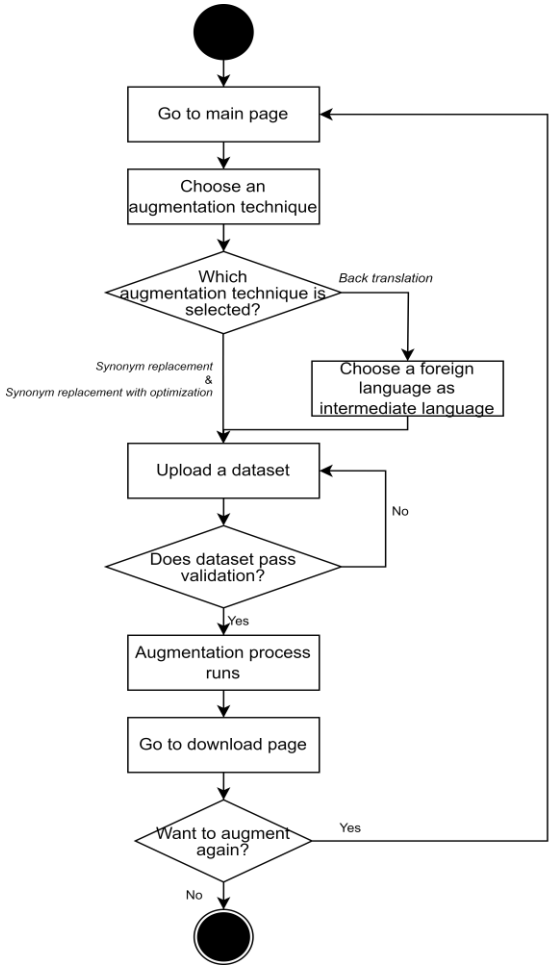

Figure 4. Activity diagram for augmentation application.

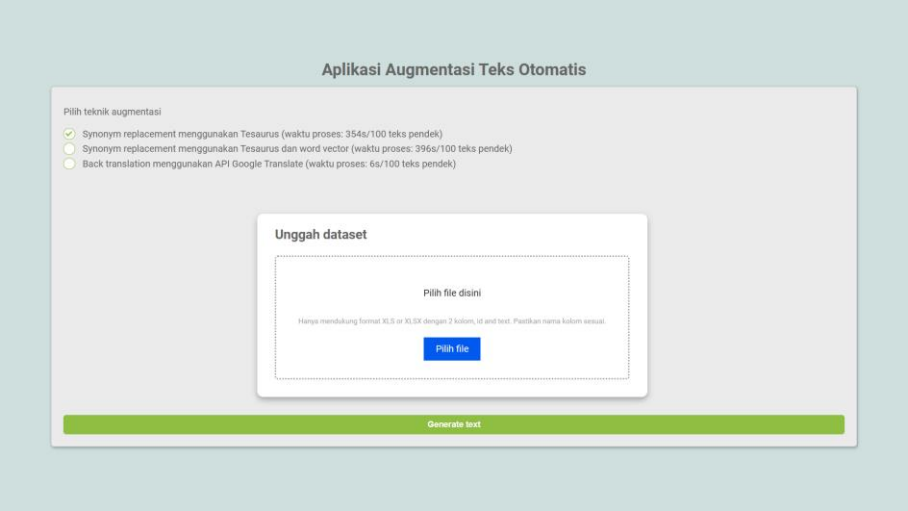

Figure 5. Main page.

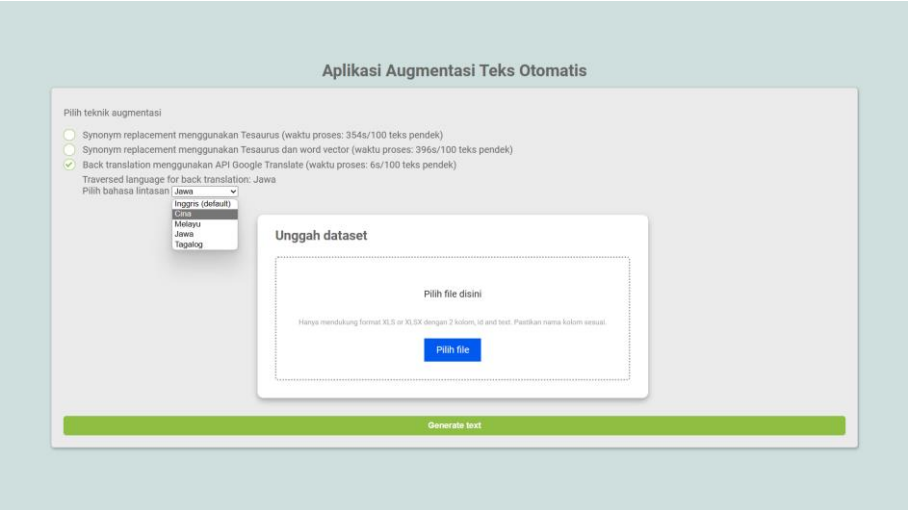

Figure 6. Interface when back translation was selected.

Then, to enter dataset file to be augmented, the user simply presses the select file button and selects the file to be uploaded. Users can only select one dataset file for augmentation. The display of the dataset upload process can be seen in Figure 7. File formats that can be uploaded are .xls or .xlsx so that the application has added file format validation to make it easier for users. Therefore, the files read by the application in the user directory are only .xls and .xlsx format files. In addition, to be able to perform augmentation, a dataset from the user is required. Therefore, the input file is set to be required by using the required attribute in HTML. When the user has not uploaded the dataset file correctly, the web application will display a pop up as a validation step in the file upload section as shown in Figure 8.

The inputted dataset file must fulfil the specified data structure so that it can be read by the application and further augmentation is performed. The dataset must at least have id and text columns with appropriate column names, no typos. Additionally, the id and text columns must not be redundant. Therefore, the web application performs strict validation of the dataset. The application will check the data structure when the generate text button is pressed and provide feedback in the form of a pop up such as validation of having inputted the dataset before. The feedback provided is error information on

the dataset. Table 1 shows the validation result feedback message for checking the dataset structure. The display regarding data structure errors in the application is shown in Figure 9.

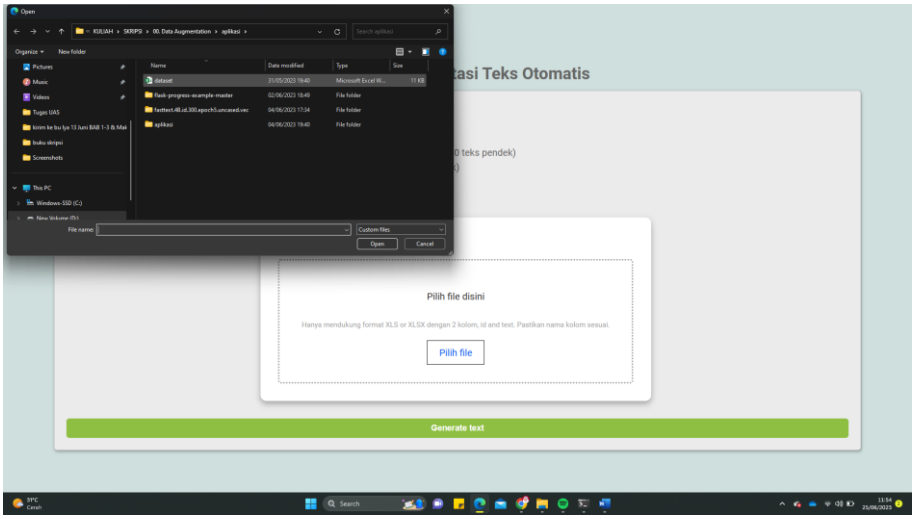

Figure 7. Dataset upload process.

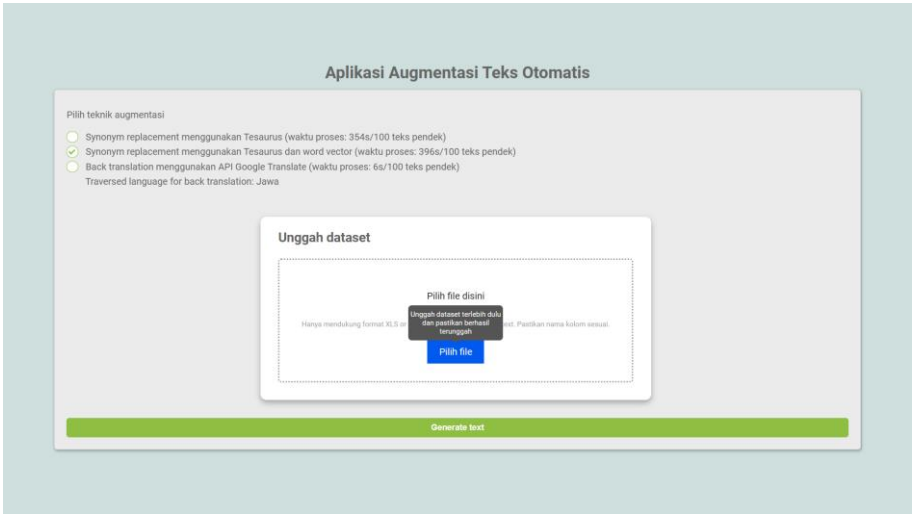

Figure 8. Dataset input validation.

Table 1. Validation messages for dataset structure.

Condition	Translated	Original (in Bahasa)
Id=0 dan text=0	Id and text columns are not detected. Make sure column exists and column name matches	Kolom id dan text tidak terdeteksi. Pastikan ada dan nama kolom sesuai
Id=0 dan text=1	Id column is not detected. Make sure column exists and column name matches	Kolom id tidak terdeteksi. Pastikan ada dan nama kolom sesuai
Id=0 dan text>1	Check again. Redundant text column and id column is not detected	Periksa kembali. Kolom text redundan dan kolom id tidak terdeteksi

Condition	Translated	Original (in Bahasa)
Id=1 dan text=0	Text column is not detected. Make sure column exists and column name is correct	Kolom text tidak terdeteksi. Pastikan ada dan nama kolom sesuai
Id=1 dan text>1	Redundant text column, only 1 column allowed	Kolom text redundan, hanya boleh 1 kolom
Id>1 dan text=0	Check again. Redundant id column and text column is not detected	Periksa kembali. Kolom id redundan dan kolom text tidak terdeteksi
Id>1 dan text=1	Redundant id column, only 1 column allowed	Kolom id redundan, hanya boleh 1 kolom
Id>1 dan text>1	Id and text columns are redundant, there can only be 1 of each	Kolom id dan text redundan, hanya boleh masing-masing 1

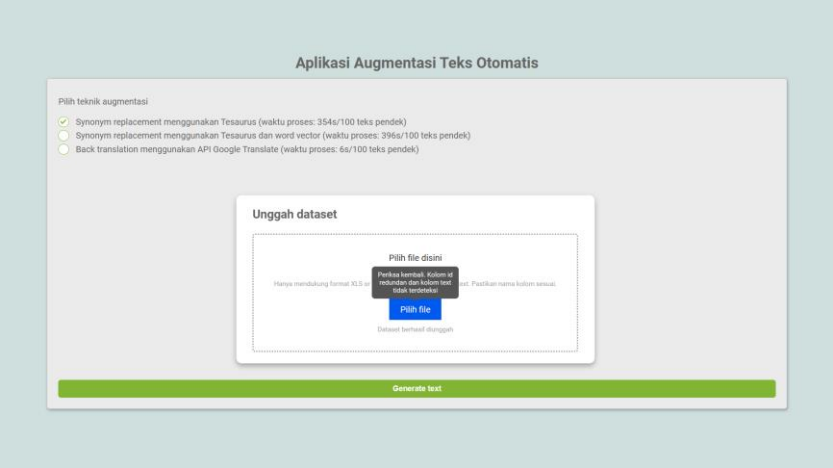

Figure 9. Validation example for dataset structure

When the generate text button is pressed and the input provided has passed validation, the web application automatically processes the dataset for augmentation. The process takes time so that web will present the generating display as information on running process as shown in Figure 10. The display apart from the title and description of generating has its transparency set with an opacity of 0.3 so that the user will not be distracted from the previous page, but can still provide information that The user needs to wait for the process that is running. The language selection input, dataset input, and the generate text button are set to readonly.

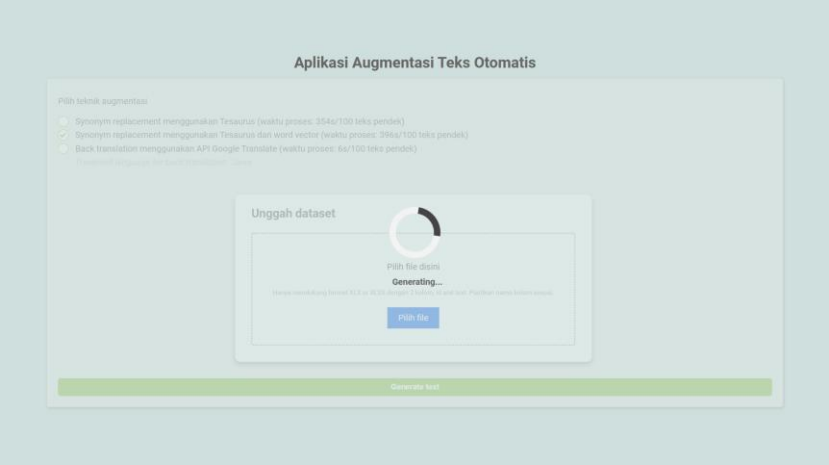

Figure 10. Processing display.

When the augmentation process is complete, the web page will automatically change to the download page as shown in Figure 21. The download page is made not much different from the main page in order to reduce the user's cognitive load. For this reason, the technique input and dataset upload sections are still displayed. Information about the selected technique and language if using back translation is still displayed to make it easier for users. However, disabled or unclickable mode is applied so that users cannot make changes. In addition, the select file button is changed to a description of the process completed which indicates the process has been successful and the button cannot be used again.

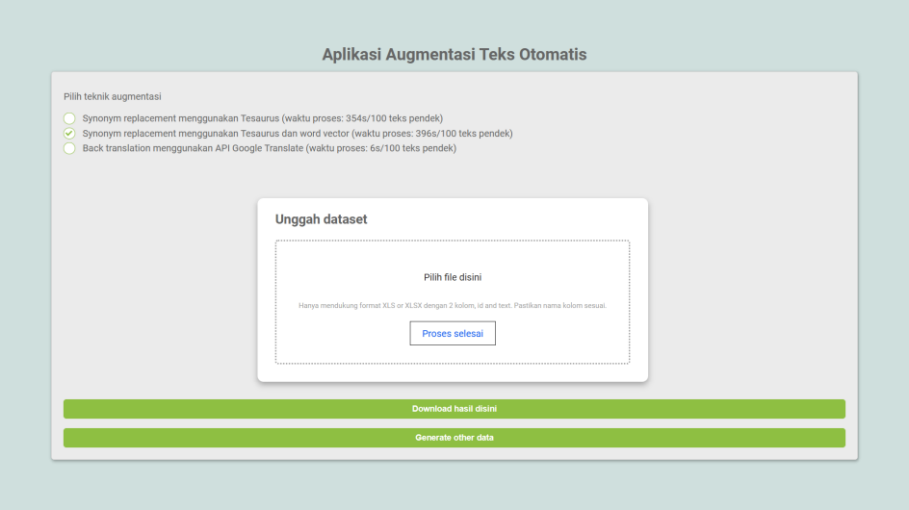

Figure 11. Download page.

To download the augmented dataset file, the user simply presses the download results button. Users can also perform augmentation again by pressing the generate other data button. This button will take the user back to the main page as when they first visited the web and can perform the automatic augmentation process from the beginning in the same way. When the user presses the download results button to download the augmented dataset, the dataset file will be automatically downloaded and added to the device's download directory.

The augmented dataset has an .xls or .xlxs format consisting of several columns. The gen_text column is an additional column from the augmentation process performed by the web application. The gen_text column contains the new augmented text derived from the original text in the text column and corresponds to the id in each instance. The other columns available in the dataset are columns from the

original dataset file. An example of the data structure of the augmentation result file from the application can be seen in Figure 12.

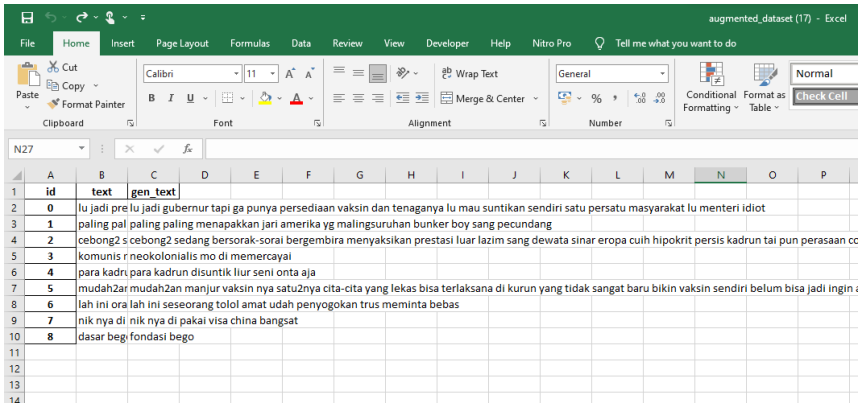

Figure 12. Application output.

3.3. Application Strengths and Limitations

This application can process text automatically to obtain synthetic text that is a paraphrase of the original text. Users can choose augmentation techniques by considering the purpose and importance as well as computation time. The augmentation performed in this application applies randomisation to the synonym replacement technique and five different intermediate language to the back translation technique, so as to produce several different synthetic texts. In addition, the augmentation process is performed on entire dataset so that users can immediately get output in the form of a new dataset. Overall, this augmentation application helps in creating new text and is easy to implement especially for handling cases of unbalanced data in text classification.

This application still needs improvement. if user wants to perform augmentation more than once, user must repeat the process as many times as desired. The development that can be done is to add input in the form of the number of outputs so that any amount of augmentation can be done in just one processing. To be widely accessible, this augmentation application requires a server for hosting and publication. Researchers only use github for documentation as well as publication.

4. Conclusion

This research has designed and built a web application for automatic Indonesian text augmentation. The application was built using the prototyping method to obtain the best results through a repetitive development and testing process. The application development process utilizes the Python Flask framework integrated with HTML, CSS, and Javascript web programming languages. The required inputs of the application are the choice of augmentation technique, the dataset, and the choice of language for back translation, while the output produced is augmented dataset file. This application can contribute to further research in performing text augmentation for Indonesians.

5. Limitations and Suggestions

There are several limitations and suggestions that can be made as improvements for further research.

1. The augmentation process can take quite a long time, depending on how large the dataset is to be augmented. Therefore, it is better if a progress bar is made that shows the progress of the augmentation process. In this study, we only used a loading display which only shows the process in progress, but does not provide information regarding progress.
2. Future research can do the same with more advanced augmentation techniques.

References

[1] [Press UGM. Big Data Untuk Ilmu Sosial: Antara Metode Riset Dan Realitas Sosial. UGM

PRESS; 2021.
[2] Chowdhary KR. Natural Language Processing BT - Fundamentals of Artificial Intelligence [Internet]. 2020. 603–649 p. Available from: https://doi.org/10.1007/978-81-322-3972-7_19
[3] Lytvyn V, Sharonova N, Hamon T, Vysotska V, Grabar N, Kowalska-Styczen A. Computational linguistics and intelligent systems. CEUR Workshop Proc. 2018;2136.
[4] Ali A, Shamsuddin SM, Ralescu AL. Classification with class imbalance problem: A review. Int J Adv Soft Comput its Appl. 2015;7(3):176–204.
[5] Sutoyo E, Fadlurrahman MA. Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Television Advertisement Performance Rating Menggunakan Artificial Neural Network. J Edukasi dan Penelit Inform. 2020;6(3):379.
[6] Verdikha NA, Adji TB, Permanasari AE. Komparasi Metode Oversampling Untuk Klasifikasi Teks Ujaran Kebencian. Semin Nas Teknol Inf dan Multimed 2018. 2018;85–90.
[7] Gu Q, Wang XM, Wu Z, Ning B, Xin CS. An improved SMOTE algorithm based on genetic algorithm for imbalanced data classification. J Digit Inf Manag. 2016;14(2):92–103.
[8] Estabrooks A, Jo T, Japkowicz N. A Multiple Resampling Method for Learning from Imbalanced Data Sets. Comput Intell. 2004;20.
[9] Sanya AD, Suadaa LH. Handling Imbalanced Dataset on Hate Speech Detection in Indonesian Online News Comments. In: 2022 10th International Conference on Information and Communication Technology (ICoICT). IEEE; 2022. p. 380–5.
[10] Rupapara V, Rustam F, Shahzad HF, Mehmood A, Ashraf I, Choi GS. Impact of SMOTE on Imbalanced Text Features for Toxic Comments Classification Using RVVC Model. IEEE Access. 2021;9:78621–34.
[11] Csányi G, Orosz T. Comparison of data augmentation methods for legal document classification. Acta Tech Jaurinensis. 2021;15(1):15–21.
[12] Lu Q, Dou D, Nguyen TH. Textual Data Augmentation for Patient Outcomes Prediction. Proc - 2021 IEEE Int Conf Bioinforma Biomed BIBM 2021. 2021;2817–21.
[13] Bugueño M, Mendoza M. Learning to combine classifiers outputs with the transformer for text classification. Intell Data Anal. 2020;24(S1):S15–41.
[14] Bock K, Garnsey SM. Language Processing. A Companion to Cogn Sci. 2008;226–34.
[15] Wilie B, Cahyawijaya S, Winata GI, Vincentio K, Li X, Kuncoro A, et al. IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding. Proc 1st Conf Asia-Pacific Chapter Assoc Comput Linguist 10th Int Jt Conf Nat Lang Process. 2020;843–57.
[16] Fahlapi R, Asra T, Kuntoro AY, Ocanitra R, Effendi L, Syukmana F, et al. Analisa sentimen vaksinasi covid-19 dengan metode support vector machine dan naïve bayes berbasis teknik smote. J Inform Kaputama. 2022;6(1):57–64.
[17] Abdurrahman, Purwarianti A. Effective use of augmentation degree and language model for synonym-based text augmentation on Indonesian text classification. 2019 Int Conf Adv Comput Sci Inf Syst ICACSIS 2019. 2019;217–22.
[18] Zhang X, LeCun Y. Text Understanding from Scratch. 2015;1–9. Available from: http://arxiv.org/abs/1502.01710.
[19] Jungiewicz M, Smywiński-Pohl A. Towards textual data augmentation for neural networks: Synonyms and maximum loss. Comput Sci. 2019;20(1):57–84.
[20] Xiang R, Chersoni E, Lu Q, Huang CR, Li W, Long Y. Lexical data augmentation for sentiment analysis. J Assoc Inf Sci Technol. 2021;72(11):1432–47.
[21] Coulombe C. Text Data Augmentation Made Simple By Leveraging NLP Cloud APIs. 2018;1–33. Available from: http://arxiv.org/abs/1812.04718.
[22] Rahma IA. Penerapan Text Augmentation untuk Mengatasi Data yang Tidak Seimbang pada Klasifikasi Teks Berbahasa Indonesia [thesis]. Jakarta (ID): STIS Polytechnic of Statistics; 2023.
[23] Dinakaramani A, Rashel F, Luthfi A, Manurung R. Designing an Indonesian part of speech tagset

and manually tagged Indonesian corpus. Proc Int Conf Asian Lang Process 2014, IALP 2014. 2014;66–9.

[24] Natasya, Girsang AS. Modified EDA and Backtranslation Augmentation in Deep Learning Models for Indonesian Aspect-Based Sentiment Analysis. Emerg Sci J. 2023;7(1):256–72.

[25] Susanto A, Meiryani. System Development Method with The Prototype Method. Int J Sci Technol Res. 2019;8(7):141–4.

[26] Carr M, Verner J. Prototyping and Software Development Approaches. Prototyp Softw Dev Approaches [Internet]. 2004;(3):1–16. Available from: https://www.google.com.my/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0CCkQFjAA&url=http://www.cb.cityu.edu.hk/is/getFileWorkingPaper.cfm?id=55&ei=73eDU6aCBo7PlAXNooGQBg&usg=AFQjCNFCEFbDyv9tNk_YuH0VpPfavJPs2A&sig2=wimyHPVpHpp.

[27] William A, Sari Y. CLICK-ID: A novel dataset for Indonesian clickbait headlines. Data Br [Internet]. 2020;32:106231. Available from: https://doi.org/10.1016/j.dib.2020.106231.