Preprocessing Pipeline of Noisy SMS Dataset

Exercise 2 – ITE Elective

Shan Hiro Rosario   
Computer Studies and Engineering  
Mandaluyong City, Philippines   
shanhiro.rosario@my.jru.edu

***Abstract*— This study explores the step by step preprocessing of a noisy dataset in preparation for Natural Language Processing (NLP) algorithms, specifically using SMS Spam Collection Dataset. It must be however, noted that the sheer volume of spam messages proves a significant challenge in natural language processing, particularly due to the specific nuances contained in the messages such as colloquial, informal, multilingual and general noisy nature of SMS messages. In observation of the publicly available dataset, the study addresses common challenges such as case normalization, removal of special characters, tokenization, stopword removal of mixed-languages such as Singlish and culturally informal fillers, and slang normalization. In regard to these unique challenges, a Standardized English is implemented by developing a custom dictionary to translate both global SMS expressions including region-specific terms, Singaporean-English in particular to further optimize algorithms and models that shall be used after the preprocessing stage has been conducted, specifically neural networks. The extraction of patterns is essential to these algorithms, particularly spam classification due to the highly meticulous methodology in cleaning of data, including but not limited to: Contribution to a practical, reproducible text cleaning pipelines for multilingual and informal SMS text-classifications.**

***Keywords*—Natural Language Processing, Python, Text Preprocessing, Multilingual Text Cleaning, Slang Normalization, Data Cleaning Pipeline**

# **Introduction**

SMS spam detection faces a significant challenge due to the informal, multilingual and noisy dataset. The complexity increases depending on the algorithm or model used in regards to these data, specifically Machine Learning and Deep Learning models. However, this paper focuses on building a preprocessing pipeline in preparation for an effective neural network training for NLP. It must be considered that the foundation of NLP consists of contextually-aware algorithms and certain datasets such as the one used in the study provides a significant amount of redundant inputs due to the nature of SMS Messages which contains informal languages, spam messages containing machine-generated texts, and ambiguous intents in regards to the cultural references. A robust pipeline that contains normalization, slang translation, stopword removal, and artifact handling such as standardization of localized expressions mixed with English or Singlish-aware cleaning through a custom distionary.

1. **RELATED WORK**

Researchers Jehad et al. (2023) developed a model consisting of Long-Short Term Memory-based classifiers for SMS Spam detection with a rigorous preprocessing pipeline with specifics includes: Tokenization, lemmatization, and stopword removal which resulted in a 98% accuracy in their model.[1]  
Uddin et al. (2024) developed and fine-tuned a transformer based models such as RoBERTA and BERT for solving spam message detection in SMS context, including the considerations for the cybersecurity domain by the use of extensive preprocessing techniques and specifically solve class-imbalance problems using text augmentation techniques[2]. Additionally, Albalawi et al. (2021) emphasizes the critical role of normalization to alleviate problems in regards to the presence of sarcastic text and multilingual data[3]. Mixed-language data provides significant challenges in NLP as explored by Winata et al.(2022) which discussed and surveyed the decades of progress over the code-switching research.[4] Researchers L. Y. Tang et al. (2024) emphasized the necessity of thorough cleaning before feature extraction methodologies such as TF-IDF on SMS spam detection model they have developed, which resulted in an 89% accuracy in a real-world dataset that outperforms the existing methods in their carefully reviewed related works. [5]

# **Methodology**

The dataset used came fromkaggle, and it provided an adequately noisy data that consists of more than 5000 rows and over 20k words. The 2 columns are specifically named as Messages and spamORham which are tagged according to being legitimate or spam. This dataset is then combined with SMS messages of the Grumbletext website of the wide-used forum in Singapore that predominantly contains Singlish or messages composed of Singaporean and English including the detailed nuances and intents of the sender. The Dataset is available in CSV file whereas the exact noise characteristics present are: typos, URLs, SMS Slang, code-switching tokens such as mixed languages of Singlish in its messages—It must be noted that these specific terms have not been discussed in the previous lectures and shall be implemented in detail below by the use of normalization and python dictionary.

Dataset Link: <https://www.kaggle.com/code/ishansoni/sms-spam-collection-dataset>

1. Removing Unwanted Columns:

A computer screen shot of a computer code

AI-generated content may be incorrect.

Figure 1: Dropping Columns

The dataset consisted of extremely noisy contents including its columns whereas the initial naming convention by the author would be preferred to be modified into Category and Message as well as dropping unnecessary column such as the first column that is unnamed and only consisted of numerical values to count each row.

1. Lowercase Values

A screenshot of a chat

AI-generated content may be incorrect.

Figure 2: Lowercase Values

It must be understood that in data preprocessing pipeline the necessary step in transforming the data is to ensure consistency which in turn allows for an efficient model to be built later on. In the figure above, we apply lowercase values on all the texts under the ‘Message’ column..

1. Removing unwanted data using Regex

A screenshot of a computer

AI-generated content may be incorrect.

Figure 3: Regex Application.

In the figure 3 above, the implementation of regex is done by removing special characters and URL in the text messages then printing out the results to observe the effectiveness of the syntax. It must be noted that removing URL's are done because we are only detecting whether the message is a spam message or a real one. The URL, i believe is outside of scope as it is only useful for phishing detection, and web scraping features

1. Stopwords List

A screen shot of words

AI-generated content may be incorrect.

Figure 4: Stopwords List

We import the required library for the available english stopwords and manually added singaporean stopwords in a dictionary format as the messages are composed of mixed singapore and english in their context. Then we added singapore\_stopwords to the variable stop\_words which contains english stopwords

1. Stopwords Removal

A screen shot of a computer code

AI-generated content may be incorrect.

Figure 5: Stopwords Removal

These are the stopword removal methodology using for loop and if else, whereas if the word is an empty space, a new line or is a word in the stop\_words dictionary, do not remove it then simply append white space using .join for better reading. At the same time, it is also a helper function, a reusable one to remove stopwords easily in coding including the one above. Then we apply the function to the entire 'Message' Column in the dataset then preview the first 10 rows that can be observed in the figure 6 below:

A screenshot of a chat

AI-generated content may be incorrect.

Figure 6: Message Content

1. Applying Tokenization  
   A computer code with text

   AI-generated content may be incorrect.

Figure 7: Tokenization

We prepare the tokenization by importing the necessary files, and simply tokenize all cleaned 'Message' using the library. In addition, print the first 10 rows to observe the result. It can be seen in the figure below:

A screenshot of a computer

AI-generated content may be incorrect.

Figure 7.1: Tokenization Output

We can immediately see above that the message are still dirty due to the fact that it contains mixed-language messages and slangs of reference in the respective cultures of the sender. To solve this conundrum, we implement a a normalization step in order to improve the implementation of algorithm later on because the messages contains a mixed of english and Singaporean with slangs on top of that. This is done in order to create a model that would generalize well.

First we create a slang dictionary. Example "Ur" is equivalent to "Your" not "You're", "Dun" is equivalent to "Do Not":. These nuances helps the algorithm better. These can be observed in the figure 8 below:

1. Slang Dictonary



Figure 8: Slang Dictionary

Since the messages consists of singaporean and english, a direct translation of the mixed messages into pure english is better for normalization. The dictionary below translates certain colloquial contractions, terms commonly used, emotional messages and slangs from singaporean language to english. These are only common terms, however, as the whole Singlish is increadible rich with terms borrowed from English, Malay, Cantonese, etc. These can be observed in the Figure 9 below:



Figure 9: Localized Singlish Dictionary

1. Normalized Message

A close-up of a computer code

AI-generated content may be incorrect.

Figure 10: Normalized Message

We create a reusable function again for normalization this time. Then apply the function to the previous tokenized message then show the results.

A screenshot of a computer

AI-generated content may be incorrect.

Figure 10.1: Output

We can observe here in the figure 10.1 that slangs, acronyms or colloquial abbreviations have been translated into clear and Standardized English without too much error except for unreadable words that the proponent of the study could not identify which exact language it came from, and with a certain degree of uncertainty of researching, it has been found out that the SMS Text messages also consists but not limited to: Indian language and references. However with a low degree of certainty, the nuances remains in the sender’s intent that, of some certain groups of young adults, gibberish texts are to be expected in their internal dialogues.

# **Results**

The Application of Lowercase preprocessing step is shown in the figure below:

A screenshot of a message

AI-generated content may be incorrect.

Figure 11: Lowercase

Sample application of removal of special characters:  
  
A screenshot of a computer screen

AI-generated content may be incorrect.

Figure 12: Special Characters Removal

Table for the number of Noise Removed will be shown in the table below:

|  |  |
| --- | --- |
| Total URLs Removed | 107 |
| Total Special Characters Removed | 24640 |
| Total Stopwords Removed | 33256 |
| Total Words Removed | 34662 |

**Table 1**

Before and after of normalization of the Message using the custom-made dictionaries in figures 8 and 9 can be seen in the figure 13 below:

A screenshot of a computer

AI-generated content may be incorrect.

Figure 13: Normalized Comparison

A screenshot of a message

AI-generated content may be incorrect.

Figure 13.1: Normalized Comparison

Removal of stopwords can be directly observed in the figure 14 below:  
A screenshot of a computer

AI-generated content may be incorrect.

Figure 14: Stopwords Removal

##### **V. Discussion**

We can directly observe the the difference between a dirty data and clean data by the figures present in the results section however, the specifics should be present in the figure 13 comparison tables of the two, namely, the differences in the abbreviations and comprehensive contents of the sentence including but not limited to “ok, joking, wif, u, oni” to “okay, joking, with, you, only” which, not only have seen an increase in efficiency in direct observation, but also for the models later on to improve their ability to generalize the results and contextually-aware of the nuances between slang of Singaporean and English languages combined.

Patterns can be extracted meaningfully when presented in such way that comprehensive sentences are immediately identified as legitimate messages as compared to spam messages consisting of unidentified combination of numerical and categorical values. For further improvements of the contextual-awareness of the development of the model further down the pipeline are not and should not be discussed in this study as it is out of the scope, whereas the focus remains solely on preprocessing textual sentences in preparation for the model building further down the line due to the meaningful patterns to be extracted from clean, coherent sentences as opposed to messages composed of irregular or mixed alphanumeric tokens, emphasizing the raw textual inputs into standardized and semantically consistent forms.

The New Noises or consisting of messages in the form of Singlish in addition to spam messages can be determined by the region-specific or cultural expressions of the SMS messages are addressed through the dictionary in figures 8 and 9, through the translation of Singaporean to English to ensure data consistency then the figures 13 shows the end result of comprehensible sentences fit to be fed to the algorithms and models. However, it must be noted that since the dictionary is custom made, certain localized expressions cannot be translated to English as the proponents of the study is not widely exposed to Singapore and its cultural nuances, therefore the ability to translate this into a more suitable structure is a significant limitation.

##### **VI. Conclusion**

The Data Preprocessing Pipeline for the SMS Spam messages of Singaporean-English contents have ben rigorously implemented using methodologies such as lowercasing, special character removal, stopword removal, tokenization, and normalization. Data Preprocessing methods are provided following the principles of preparation for model building or algorithm development. The study showcased efficient method of structuring incomplete and mixed-language words into a coherent and contextually-aware sentences due to the dictionary implementation. A thoroughly cleaned dataset improves the efficacy of machine/deep learning models in its ability to generalize well.

Furthermore, the complexity of NLP tasks, in addition to the understanding of the key concepts of the models, the most challenging aspects are present in the data preprocessing step as the aim of the course subject is not to build an algorithm but to understand how to effectively use it. Therefore, considerations should be given to this stage as it directly affects both the algorithm and the data analyst using the algorithm.

Additional research should be given to Lemmatization, which is follows concepts of reducing a word to its base or dictionary form, called lema which helps greatly with Text normalization, reducing vocabulary size and improving meaning retention, or in other words, makes the algorithm later on be language-aware whereas the word “Better” would be identified as “good” rather than producing simple output such as “Bett” or other case examples.

##### VII. References

[1] M. A. Jehad, A. Bou Nassif, and M. AlShabi, “Spam SMS detection based on long short-term memory and recurrent neural network,” in Proc. SPIE 12538, Data Mining, Machine Learning, and Artificial Intelligence, vol. 12538, May 2023, Art. no. 125381Y. [Online]. Available: https://doi.org/10.1117/12.2674147

[2] M. A. Uddin, M. M. Rahman, A. Sarker, M. H. Wahid, and A. Wahid, “ExplainableDetector: Exploring transformer-based language modeling approach for SMS spam detection with explainability analysis,” arXiv preprint arXiv:2405.08026, May 2024. [Online]. Available: https://arxiv.org/abs/2405.08026

[3] S. Albalawi, J. Buckley, and N. Nikolov, “A systematic literature review on spam content detection and classification,” Expert Systems with Applications, vol. 169, pp. 114395, Mar. 2021. [Online]. Available: https://doi.org/10.1016/j.eswa.2020.114395

[4] G. I. Winata, Z. Lin, and P. Fung, “The decades progress on code-switching research in NLP: A systematic survey on trends and challenges,” arXiv preprint arXiv:2212.09660, Dec. 2022. [Online]. Available: https://arxiv.org/abs/2212.09660

[5] L. Y. Tang, F. H. Ali, and N. A. M. Isa, “A complex-network based model for SMS spam filtering,” Computer Networks, vol. 255, pp. 110148, Dec. 2024. [Online]. Available: https://doi.org/10.1016/j.comnet.2024.110148

‌

‌