e-ISSN: 2686-2573
DOI: 10.21512/emacsjournal.v7i2.13503

The Impact of Text Preprocessing in Sarcasm Detection on Indonesian Social Media Contents

Nicholaus Hendrik Jeremy

Computer Science Department, School of Computer Science,
Bina Nusantara University,
Jakarta, Indonesia 11480
nicholaus.jeremy@binus.ac.id

Correspondence: nicholaus.jeremy@binus.ac.id

Abstract – *Sarcasm is a way to convey something but delivered in the opposite way. This behavior is common on social media, where there are plenty of examples. On natural language processing, the task on its own is difficult primarily due to the lack of context. To add another layer of difficulty, communication in social media is done colloquially. One sacrasm benchmark, IdSarcasm, has alleviated one key issue in the development of sarcasm detection. However, there has not been an attempt to further preprocess the input before feeding them into the model. Pre-trained language models always use preprocessed corpus to ensure that the model is built upon quality dataset. Based on the current condition of IdSarcasm, further preprocessing step is necessary to ensure better quality. Specifically, the additional steps needed are handling HTML code, code-mixing, and colloquial writing which consists of shortened form, extended form, spelling variation, and reduplication. Several scenarios are created to observe the effect of additional preprocessing steps. Each additional preprocessing step is also tested to observe the effect of the preprocessing step independently. We prove that preprocessing step is still prevalent for data sourced from social media, and we recommend IndoNLU's IndoBERT or large multilingual model to be used for sarcasm classification.*

Keywords: *Sarcasm Detection; IdSarcasm; Social Media; Preprocessing; Natural Language Processing*

I. INTRODUCTION

Sarcasm is defined as a way of communicating a meaning by saying the opposite (Zhu & Wang, 2020). The aim of this contradiction is to show critics or putting the communicator in the aggresive stance (Toplak & Katz, 2000). While the sarcasm itself is just a way of constructing a sentence, its characteristics are heavily influenced by verbal cues (Caucci & Kreuz, 2012). As such, to understand sarcasm is an ability (Glenwright & Pexman, 2010). However, in the context of text-based social media, those cues does not exist, making sarcasm detection significantly challenging.

The research on sarcasm has been growing steadily. For niche topic, in this case on Indonesian dataset, there has been numerous attempts to improve the quality of sarcasm detection. One pivotal research in this topic is done by providing a public benchmark named IdSarcasm (Suhartono et al., 2024) to the masses. According to the experiment, under F1-Score the best result yielded was 62.7% on Reddit dataset and 76.9% on Twitter dataset. The potential issues found within the research are the lack of broader context, lack of transparency in data used on each of previous' sarcasm detection research, and the domain that is restrictred to social media only. While the first issue is true (Eke et al., 2021; Helal et al., 2024; Wang et al., 2015) in order to achieve better prediction result, the rest of it is not of a significant issue. The research also overlook the preprocessing step, which is crucial in building a dataset.

Pre-trained model is trained on curated dataset. This includes quality filtering, de-duplication, privacy reduction, and tokenization (Matarazzo & Torlone, 2025). However, scrutinized preprocessing step needs to be also done if the source of the dataset allows for more colloquial writing. For example, large part of trained dataset for IndoBERT (Wilie et al., 2020) is OSCAR (Suárez et al., 2020), which while is stated as a mix of formal and colloquial, the majority of the data is written cleanly. BERT (Devlin et al., 2019) itself is trained on clean-format data . This in turn raises the importance of preprocessing the text to ensure that the target task data follows the quality of the base data.

Therefore, the contribution of the research can be stated as follows:

- Providing the continuation of data preprocessing step to improve the quality of the benchmark
- Providing the comparison of before and after preprocessing step to highlight the important part of preprocessing that is required to be done.
- Open up the possibility of overseeing the benchmark and assess further which process can be scrutinized

Several research on Indonesian sarcasm detection has been mentioned briefly on previous section. This section will further explore the findings of each paper and make highlights of the preprocessing step, if there is any.

Proposed technique done by Rahayu et al. (2018) utilizes two-step classification in which positive sentiment tweets from twitter are then classified again to positive/sarcasm label, resulting in a roughly equal 1:1:1 positive:sarcasm:negative ratio. When classifying sarcastic tweets, it is also noted that a combination TF-IDF and cosine similarity as a feature outperforms Bag of Words (BoW). The preprocessing steps done in this research are casefolding, filtering, stemming, interjection removal, and punctuation removal. Since the model used are classical, namely Naïve Bayes and k-NN, the usage of TF-IDF also adds another layer of stopword removal which is a norm when working with text which will be fed to classical models (Dolamic & Savoy, 2010; Sarica & Luo, 2021). This, in turn, makes the model perform better, as proven that the model runs on BoW feature achieved 50% F1-Score while the model runs on TF-IDF and cosine similarity features achieved 82%.

Technique used by Alita et al. (2019) highlights the importance of conveyed emotion written in the tweet from twitter. This includes counting the occurrence of emotion-based tokens (including the conversed emoticons), counting the words that are sentiment-based, counting the number of hashtags, and counting the number of several punctuations. The sarcasm detection is done under two-step classification much like the previous research. The experiment concluded that using emotion-based token, specifically emoticons, only slightly improves the accuracy but not F1-Score. According to the list of features used, it can be argued that since positive and sarcastic tweets shared most of the similarity, the classification of sarcastic-only tweets can be significantly challenging, and there is more nuance when it comes to emotion than emoticons.

Novel approach by Rosid et al. (2024) combines convolution, multi-head attention, and bi-directional gated recurrent unit which is proposed as MHA-CovBi model. The tweets are code-mixed, meaning that for a tweet it can have multiple languages (in this case Indonesia-English). Their proposed model able to perform on significant advantage, yielding 94.38% F1-Score on their own dataset. However, their performance on other dataset (Misra & Arora, 2023) can't outperform the performance (bi-directional LSTM with attention mechanism and convolution) on accuracy, with only 1.58% difference. On their preprocessing, the treatments done consist of filtering, stopword removal, casefolding, tokenization, lemmatization, non-ASCII character removal, and slang conversion. On top of that, a group of auxiliary features are generated by counting the number of dots, exclamation marks, question marks, extended words, and uppercase letters. These auxiliary features able to leverage model performance by 1.51%. As for code-mixing, Indonesian token is processed using FastText, English token is processed using GloVe, and non-identifiable or non-Indonesian/English token is translated to English before processed accordingly.

Another approach using neural model, speicifically LSTM (Khotijah et al., 2020), utilizes Paragraph2Vec to obtain context within the tweets. The model able to achieve 95.19% F1-Score under balanced dataset and 97.31% under imbalanced dataset. However, upon testing under new dataset, which is not split from the training set, the performance drops significantly to 87.03% and 68.18% respectively. This shows the significant role in keeping the data balanced to prevent overfit, although it is rarely possible. As for the preprocessing step, the treatments done are normalization, casefolding, stopword removal, and sentence reversal.

II. METHODS

A continuation of preprocessing step on IdSarcasm is provided in this research which is done after the preprocessing step that has been done. The new flow when working with the dataset is therefore changed insignificantly, ensuring that any work done using the dataset does not need to overhaul the code to be repaired. The research will be done on Reddit's dataset of IdSarcasm and not attempted on Twitter dataset to comply with the platform's agreement and policy[1] . The continuation of preprocessing steps will be discussed under the following subsections.

2.1 On Unparsed HTML Character Entities

HTML character entities are used for several special characters, notably reserved HTML characters. HTML character entities may appear within scraped data. Specifically for this dataset, entities from XML 1.0 will be converted to the respective symbols which are:

[1] See https://developer.x.com/en/developer-terms/agreement-and-policy under "Content redistribution". The original paper does not provide the approval from the platform, hence the hesitation.

- & for ampersand (&)
- < for less than (<)
- > for greater than (>)
- ' for single-quote or apostrophe-quote (‘)
- " for double-quote or quotation mark (“)

2.2 On Code-Mixed Sentences

Code-mixing is usually done by a bilingual person, switching language within the same sentence. Code-mixed sentences also occur on Reddit dataset, meaning that it’s possible for them to be added as a new vocabulary rather than treating it as an existing token. This may create sparsity when developing the vocabulary, and some might be truncated depending on the hyperparameter setting. To prevent this, all sentences is translated to Indonesia. This process is done with the help of automation.

2.3 On Colloquial Writing in Social Media

On social media, colloquial writing is the norm along with text (Al Shlowiy, 2014; Kemp, 2010). Both writing styles are popular as it allows faster typing and rapid communication, especially on smaller devices such as smartphones. The components that constitute such writing style include:

2.3.1 Shortened Form

Shortened form consists of three subcategories, namely abbreviation, acronym, and initialism. While standard shortened form can be seen on formal writing (e.g. institution, organization, title), textism has its own shortened forms. In Indonesia, some notable shortened forms can be seen in table 1.

Table 1. Examples of Shortened Form

Actual word	Shortened form
yang (which)	yg
tidak (no)	tdk
ya (yes)	y
kali (maybe, perhaps)	x
nggak (no)	g, ngga, ga, gk, gak
nggak bisa (cannot)	gbs
kalau (if)	kl
bisa (can)	bs
sekarang (now)	skrg
langsung (immediate/ly)	lgsg
gue (i)	gw
tapi (but)	tp
bukan (not)	bkn
sudah (already)	udh, udah, dah
itu (that)	tu
ini (this)	ni, nih
dengan (with)	dgn
pakai (using)	pk, pake

2.3.2 Extended Form

Extended form is used to emphasis on a certain word or to mimic prolongation that is not caused by stuttering. In text form, this can be done by extending the usage of a certain character to an extreme. To mitigate this, all sequences of consecutive characters are limited to two. This includes group of repeated characters, such as “hahaha” or “wkwkwk” (Indonesian laugh).

2.3.3 Spelling Variation

Spelling variations occur especially in informal conversation. This can happen due to dialect, loan word, or collective societal habit (Devianty, 2021). Several examples of spelling variation can be seen in table 2. While it is debatable that converting variation might hurt the diverse writing style, since sarcastic utterance depends on the semantic, conversion is safe to apply. However, to preserve the variety, any informal form will be preserved, and the abbreviation therefore will be expanded to the respective form. Some spelling variations may also fall under shortened form category.

Table 2. Examples of Spelling Variations

Actual word	Spelling Variation(s)
gue (i)	gw, guwe
nggak (no)	g, ngga, ga, gk, gak
lu (you)	loe, elu
cowok (boy)	cowo
cewek (girl)	cewe
bosan (bored)	bosen
malas (lazy)	males
benar (correct)	bener
seram (scary)	serem
siram (to water)	sirem
anjir (dog, exclamation)	njir, jir, anjim
tahu (know)	tau
bagaimana (how)	gmn, gimana
gue (i)	gw, guwe
nggak (no)	g, ngga, ga, gk, gak
lu (you)	loe, elu
cowok (boy)	cowo
cewek (girl)	cewe

2.3.4 Reduplication

Reduplication (kata ulang) is common in Indonesia, for example jalan-jalan (travel/hang out), hati-hati (be careful), and malam-malam (at night). To shorten the typing process, the usage of number 2 (e.g. jalan2) is often used, either as is or written using superscript (e.g. jalan2). Alternatively, a double quote at the end of the word can also be used (e.g. jalan”).

Table 3. Examples of how each additional preprocessing step works. Changes are represented by bold and underline

Scenario #	Original	Processed
3	**It's better** daripada mau masukan eskrim tapi bau rendang? Eskrim rasa rendang?	**Lebih baik** daripada mau masukan eskrim tapi bau rendang? Eskrim rasa rendang?
4	**>**I guess kayaknya Perhitungan Pajak mesti diajarin sejak SMA atau akhir kuliah	**>**I guess kayaknya Perhitungan Pajak mesti diajarin sejak SMA atau akhir kuliah
5	**Krn** nembusin golongan penguasa mungkin, jadi ada konflik pun bukan dianggap **sbg** konflik agama melainkan konflik antar kerajaan dan rakyat nya	**Karena** nembusin golongan penguasa mungkin, jadi ada konflik pun bukan dianggap **sebagai** konflik agama melainkan konflik antar kerajaan dan rakyat nya
6	ibu ceo bales wa saya **ibuuu** :(nanti kalo udah besok saya diomelin lagi kenapa gak dari hari ini padahal udah diwa dari kapan **tauuuu** :(	ibu ceo bales wa saya **ibuu** :(nanti kalo udah besok saya diomelin lagi kenapa gak dari hari ini padahal udah diwa dari kapan **tauu** :(
7	Lagunya **rasa2** 'warkop dki'	Lagunya **rasa-rasa** 'warkop dki'

2.4 On Other Common Preprocessing

There are various other preprocessing steps that work on several other research, however under consideration that the model that will be inspected upon is pre-trained model, consideration must be taken:

- *Casefolding*: in this research, casefolding is done to all lowercase through the tokenizer of the respective language model.
- *Filtering*: this step has been applied to IdSarcasm.
- *Stemming/Lemmatization*: with the usage of word-piece tokenizer, these techniques become obsolete.
- *Stopword removal*: recent language models can accommodate stopwords which potentially contribute to enriching the context of the sentence, therefore it is not done.

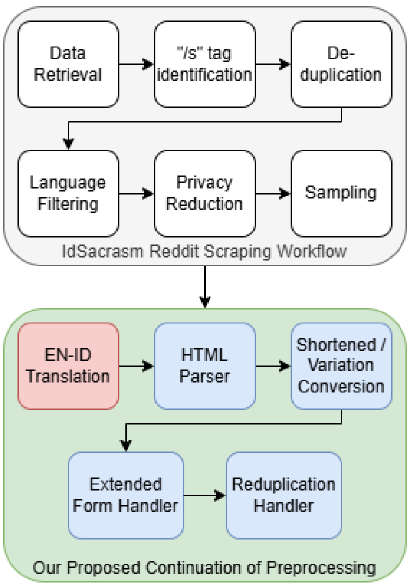

Figure 1. Additional preprocessing step workflow

The flow of the process can be seen in Figure 1. The step in red signifies that the placement or order of the said steps matter, while the step in blue signifies that the placement or order of the said steps does not. To explore the impact of each step, isolation preprocessing is also done by only doing one of the steps within the continuation boundary. In total there will be 7 data variations. Examples of each step are shown in Table 3. The specifications and the number representing the variation are as follows:

1. The entire process
2. Without translation
3. Translation only
4. HTML parsing only
5. Shortened/variation conversion only
6. Extended form handler only
7. Reduplication handler only

Evaluation is done under F1-Score, as the data suffers from imbalance and other metric fails to represent the performance properly. The models used are IndoBERT from IndoLEM, IndoNLU, mBERT, and XLM-R. The elimination of traditional machine learning models and zero-shot approach is due to their poor performance on the original paper. Arguably these selected models should be able to represent the changes of performance under other unselected models. The reported value is obtained from running validation and test set under the best performing model.

III. RESULTS AND DISCUSSION

The results are presented under two tables, Table 4 for validation result and Table 5 for test result. For the sake of brevity, IndoBERT is implied under both IndoNLU and IndoLEM and the size used is written as subscript.

Results from both table show that model trained on untreated data is outperformed by model that is trained on treated data. This implies that data treatment is still necessary to further improve the model performance. The rate of significance,

Table 4. F1 Result for Validation. IndoNLU and IndoLEM are IndoBERT models

Scenario	IndoNLU$_{BASE}$	IndoNLU$_{LARGE}$	IndoLEM$_{BASE}$	mBERT	XLM-R$_{BASE}$	XLM-R$_{LARGE}$
Baseline	0.587	0.608	0.581	0.550	0.609	0.627
Scenario 1	0.583	0.607	0.581	0.563	0.622	0.593
Scenario 2	0.590	**0.615**	**0.589**	**0.604**	0.617	0.618
Scenario 3	0.583	0.609	0.583	0.516	0.587	0.609
Scenario 4	0.592	0.602	0.587	0.549	0.605	0.624
Scenario 5	**0.602**	0.609	0.583	0.585	0.627	0.599
Scenario 6	0.585	0.608	0.576	0.563	**0.634**	0.614
Scenario 7	0.591	0.610	0.557	0.579	0.608	**0.633**

Table 5. F1 Result for Test. IndoNLU and IndoLEM are IndoBERT models

Scenario	IndoNLU$_{BASE}$	IndoNLU$_{LARGE}$	IndoLEM$_{BASE}$	mBERT	XLM-R$_{BASE}$	XLM-R$_{LARGE}$
Baseline	0.588	0.598	0.573	0.555	0.600	0.607
Scenario 1	0.585	0.607	0.578	0.516	0.585	0.604
Scenario 2	0.576	0.605	0.561	0.550	0.594	0.618
Scenario 3	**0.599**	0.596	0.581	0.515	0.561	0.603
Scenario 4	0.589	0.596	0.578	0.528	0.556	0.593
Scenario 5	0.592	**0.609**	0.562	**0.556**	**0.613**	0.612
Scenario 6	0.593	0.609	**0.581**	0.539	0.593	0.607
Scenario 7	0.583	0.602	0.576	0.533	0.575	**0.626**

however, may vary. The result also shows that during validation, scenario 2 in which all but translation is being applied generates the most amount of model that performs better. We argue that the quality of the automated translation plays a huge role in this regard, as proven that under multilanguage model this approach instead degrades the performance. The test result table, instead, shows that scenario 5 in which only shortened/variation conversion step is being done achieves the most amount of model with improvement. This reinforces the idea that variation in writing under colloquial writing may damage the vocabulary within the embedding, thus reducing the performance.

Further discussion to explore the significance of each step within the newly proposed continuation is needed. Specifically, there needs to be analysis of scenario 3 through 7.

We have mentioned for scenario 3 that the tendency for the model's performance is leaning towards no change to reduction of performance, potentially caused by the lack of proper translation done by automation. We also find that consistently the model that are performing worse using translated corpus are multilingual model. This hints that sarcasm delivery in English is significant in Indonesia's internet culture and cannot be replaced merely with its translation, potentially changing the meaning and losing its sarcastic value.

For scenario 4, we find changing HTML does not change much, with the exception of mBERT and XML-RBASE, which both are smaller-sized multilanguage model. To explore the possibility of why, we need to go back to how tokenizer works. Non-converted HTML character entities have the format of <&XXX;>, which begins and ends with special character. It is possible for the tokenizer to remove the said special character and obtain the word within the character entities. In Reddit, especially, one special character that is often used is the symbol greater than “>”, to indicate restating. Often this is done by user to add direct comment to the said statement, potentially in sarcastic way. Since the letters within the character entities got trained, it is possible that this affect the model's performance, since it can be included within the vocabulary as well due to the frequency.

Model's performance trained on Scenario 5 tends to have its performance increased. We argue that this correlates to how vocabulary is created, which is through frequency. Proven through all monolingual model's performance increased, we argue that limiting word variance allows for other words that might not be as significant frequency-wise to be included within the vocabulary. This, in turn, allows for more words learned without increasing the vocabulary size. While the concept of modifying the word to increase its frequency can be recalled in lemmatization/stemming, this is different as: (1) usually the suffixes are preserved, and (2) this occurs more often in stopwords.

Scenario 6's result can be explained through going back again to the tokenizer. BERT-based model tokenizes group of characters, for example “hahaha” to “ha, ##ha, ##ha” and “wkwkwk” to “w, ##k, ##w, ##k, ##w, ##k”. Since this extension can go beyond the example provided, the frequency of the respective token will increase rapidly. This, in turn, affects the vocabulary created. We argue that most likely these extended form occurs often in either sarcastic and non-sarcastic posts, and reduces the performance since this occupies the place in vocabularies. Not to mention that extended form can be a random sequence of letters or inconsistently repeating characters (e.g. a randomized sequence of “w” and “k” such as “wwkwkkwkw” can often be found).

Laslty, models trained on data treated with scenario 7 have inconclusive result. For one, only IndoNLU’s IndoBERT model gains an increase in performance, but not in IndoLEM’s. IndoLEM’s IndoBERT is further trained on formal-writing data, such as wikipedia and electric newspaper (summarization dataset is obtained from Liputan6). This potentially desaturating the knowledge trained from the initial learning of IndoNLU which includes corpus from Twitter. Since training data used by IndoLEM has no innate sarcasm, this potentially reduces the final performance. The reduction also happens on multilingual model, with the exception of XLM-RLARGE. We argue that larger parameter allows for more context learned.

IV. CONCLUSION

This research presented the continuation of the preprocessing step done to IdSarcasm to emphasize the importance of data quality within the training scheme of predictive model. While not all scenarios significantly increase the performance of the model, the fact that all models benefit from the additional scenarios proves that the continuation of the preprocessing step is still necessary. Future creation of dataset should keep this information in mind.

For monolingual model in Bahasa Indonesia, IndoNLU’s IndoBERT is preferred over IndoLEM when it comes to sarcasm detection, as their performance is better. For multilingual, the larger the better to allow more context learned, as different languages possess unique grammatical characteristics and ways to convey sarcasm.

AUTHORS’ CONTRIBUTIONS

Conceptualization, Data curation, Formal analysis, Methodology, Validation, Visualization, Writing: NHJ

OPEN DATA

The dataset used can be accessed through the following:*https://huggingface.co/datasets/w11wo/reddit_indonesia_sarcastic*

REFERENCES

Al Shlowiy, A. (2014). Texting abbreviations and language learning. International Journal of Arts & Sciences, 7(3), 455.

Alita, D., Priyanta, S., & Rokhman, N. (2019). Analysis of Emoticon and Sarcasm Effect on Sentiment Analysis of Indonesian Language on Twitter. Journal of Information Systems Engineering and Business Intelligence, 5, 100. https://doi.org/10.20473/jisebi.5.2.100-109

Caucci, G. M., & Kreuz, R. J. (2012). Social and paralinguistic cues to sarcasm. Humor, 25, 1–22. https://doi.org/10.1515/humor-2012-0001

Devianty, R. (2021). The Use of Standard and Unstandardised Words in Indonesian (Penggunaan Kata Baku Dan Tidak Baku Dalam Bahasa Indonesia). EUNOIA (Jurnal Pendidikan Bahasa Indonesia), 1, 121. https://doi.org/10.30821/eunoia.v1i2.1136

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North, 4171–4186. https://doi.org/10.18653/v1/N19-1423

Dolamic, L., & Savoy, J. (2010). Brief communication: When stopword lists make the difference. Journal of the American Society for Information Science and Technology, 61(1), 200–203. https://doi.org/10.1002/asi.21186

Eke, C. I., Norman, A. A., & Shuib, L. (2021). Context-Based Feature Technique for Sarcasm Identification in Benchmark Datasets Using Deep Learning and BERT Model. IEEE Access, 9, 48501–48518. https://doi.org/10.1109/ACCESS.2021.3068323

Glenwright, M., & Pexman, P. M. (2010). Development of children’s ability to distinguish sarcasm and verbal irony. Journal of Child Language, 37, 429–451. https://doi.org/10.1017/S0305000909009520

Helal, N. A., Hassan, A., Badr, N. L., & Afify, Y. M. (2024). A contextual-based approach for sarcasm detection. Scientific Reports, 14. https://doi.org/10.1038/s41598-024-65217-8

Kemp, N. (2010). Texting versus txtng: Reading and writing text messages, and links with other linguistic skills. Writing Systems Research, 2, 53–71. https://doi.org/10.1093/wsr/wsq002

Khotijah, S., Tirtawangsa, J., & Suryani, A. A. (2020). Using LSTM for Context Based Approach of Sarcasm Detection in Twitter. ACM International Conference Proceeding Series. https://doi.org/10.1145/3406601.3406624/SU

PPL_FILE/A19-KHOTIJAH-CORRIGENDUM.PDF
Matarazzo, A., & Torlone, R. (2025). A Survey on Large Language Models with some Insights on their Capabilities and Limitations.
Misra, R., & Arora, P. (2023). Sarcasm detection using news headlines dataset. AI Open, 4, 13–18. https://doi.org/10.1016/J.AIOPEN.2023.01.001
Rahayu, D. A. P., Kuntur, S., & Hayatin, N. (2018). Sarcasm detection on Indonesian twitter feeds. International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), 2018-October, 137–141. https://doi.org/10.1109/EECSI.2018.8752913
Rosid, M. A., Siahaan, D., & Saikhu, A. (2024). Sarcasm Detection in Indonesian-English Code-Mixed Text Using Multihead Attention-Based Convolutional and Bi-Directional GRU. IEEE Access. https://doi.org/10.1109/ACCESS.2024.3436107
Sarica, S., & Luo, J. (2021). Stopwords in technical language processing. PLoS ONE, 16. https://doi.org/10.1371/journal.pone.0254937
Suárez, P. J. O., Romary, L., & Sagot, B. (2020). A Monolingual Approach to Contextualized Word Embeddings for Mid-Resource Languages. Proceedings of the Annual Meeting of the Association for Computational Linguistics, 1703–1714. https://doi.org/10.18653/v1/2020.acl-main.156
Suhartono, D., Wongso, W., & Tri Handoyo, A. (2024). IdSarcasm: Benchmarking and Evaluating Language Models for Indonesian Sarcasm Detection. IEEE Access, 12, 87323–87332. https://doi.org/10.1109/ACCESS.2024.3416955
Toplak, M., & Katz, A. N. (2000). On the uses of sarcastic irony. Journal of Pragmatics, 32, 1467–1488. https://doi.org/10.1016/S0378-2166(99)00101-0
Wang, Z., Wu, Z., Wang, R., & Ren, Y. (2015). Twitter sarcasm detection exploiting a context-based model. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9418, 77–91. https://doi.org/10.1007/978-3-319-26190-4_6
Wilie, B., Vincentio, K., Winata, G. I., Cahyawijaya, S., Li, X., Lim, Z. Y., Soleman, S., Mahendra, R., Fung, P., Bahar, S., & Purwarianti, A. (2020). IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding. Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, 843–857. https://doi.org/10.18653/v1/2020.aacl-main.85
Zhu, N., & Wang, Z. (2020). The paradox of sarcasm: Theory of mind and sarcasm use in adults. Personality and Individual Differences, 163. https://doi.org/10.1016/j.paid.2020.110035