

IndoBERTweet for Sarcasm: Evaluating Domain-Adapted Transformers for Indonesian Twitter Sarcasm Classification

Fitrahtur Rahman, Abba Suganda Girsang

Computer Science Department, Bina Nusantara University, Jakarta, 11480, Indonesia

fitrahtur.rahman@binus.ac.id

Abstract. This study investigated deep learning approaches for detecting sarcasm in Indonesian Twitter data. Fine-tuned transformer models including IndoBERTweet, IndoBERT, and IndoGPT were evaluated on a dataset of 8700 tweets and compared to an LSTM baseline. IndoBERTweet achieved the best performance with an F1-score of 89.11%, demonstrating the promise of domain-adapted models for this task. The findings provide useful insights into effective sarcasm classification strategies for low-resource languages. Further research on integrating contextual features is warranted to address limitations.

Keywords: sarcasm detection, natural language processing, Transformer, BERT, Twitter

1. Introduction

In recent years, Twitter usage in Indonesia has seen a considerable uptick, making it a vital platform for social discourse and public opinion mining. Indonesia ranked fifth globally for the most Twitter users with approximately 18.45 million users As of January 2022 (Dixon, 2022), contributing to an array of topics from politics to daily social issues in the form of text. The growing number of users presenting an opportunity to understand public sentiments on various issues, including the nuanced linguistic feature of sarcasm.

However, text-based communication lacks non-verbal cues. It does not incorporate supporting factors such as facial expressions, physical interaction, and auditory (Lieberman & Schroeder, 2020). The informal language used on Twitter, as well as the 280-character limit, make it difficult to discern subtleties on sarcasm. As a form of language, sarcasm often communicates the opposite of what is intended. This disparity is used to show dissatisfaction with a preceding proposition, frequently in the form of disdain (Oprea & Magdy, 2019). The use of sarcasm can be directed at individuals or groups.

Previous research on classifying Indonesian tweets has predominantly used traditional machine learning models and manual hyperparameter tuning. This paper introduces a deep learning-based approach to sarcasm detection, specifically employing the domain-adapted IndoBERTweet. Leveraging its self-attention mechanism, the model can capture contextual meaning between words simultaneously and thereby better understand the intricate patterns that often indicate sarcasm in textual data. To further refine the model, we incorporated automated hyperparameter tuning using Asynchronous Successive Halving and Tree-Structured Parzen Estimator algorithms to optimize the fine-tuning phase. The effectiveness of our proposed method was evaluated by comparing it with baseline LSTM model and two other pre-trained models, IndoGPT and IndoBERT.

2. Literature Review

2.1. Sarcasm Detection

The domain of sarcasm detection in social media, notably on Twitter, has experienced a rapid proliferation of methodologies, which can be generally clustered into three major approaches: traditional machine learning models, deep learning-based architectures, and ensemble or hybrid techniques. Each cluster leverages unique features and techniques that contribute to the sophistication of the sarcasm detection task.

Studies such as those by Rahayu et al. (2018), Yunitasari et al. (2019), and Eke et al. (2021) are representative of the efforts to use traditional machine learning algorithms for sarcasm detection. Rahayu et al. experimented with combinations of Bag of Words and Naïve Bayes, as well as TF-IDF and k-Nearest Neighbor, finding that the latter combination outperformed the former with an F-measure of 82% in Indonesian Twitter dataset. Yunitasari et al. also delved into feature engineering, utilizing unigrams and feature sets from Bouazizi & Ohtsuki (2017) to improve sentiment analysis by 5.49% using their Indonesian Twitter dataset. They employed a Random Forest classifier, showcasing that traditional classifiers are still effective for this task. Eke et al. pushed the boundaries of feature engineering by adopting a two-stage Multi-feature Fusion Framework and employing a plethora of classifiers and feature extractions. Their work accentuated the importance of feature selection, such as Pearson correlation, to achieve a high precision score of 0.947 with a Random Forest classifier.

Deep learning methods have emerged as a formidable approach for sarcasm detection, as evidenced by the works of Ren et al. (2020), Ashok et al. (2020), and Dutta & Mehta (2021). Ren's Multi-level Memory Network (MMNSS) uses a double-layered architecture to capture sentiment semantics specific to sarcasm through LSTM encoder layers and to discern contrasting sentiments through Local-max Convolutional Neural Networks (LM-CNN). Ashok's model incorporated BERT into its embedding layer, offering the advantage of transfer learning, and fine-tuned an LSTM network with genetic algorithms for hyperparameter optimization. Dutta's work involved the use of a fusion method, C-RNN,

that combined CNN and Bi-LSTM layers, achieving an accuracy rate of 84.73% on a Kaggle dataset (Misra, 2019). These studies emphasize the power of deep learning architectures, especially when used in tandem with advanced optimization techniques.

The necessity for complex models that capture the intricacies of sarcasm has given rise to ensemble methods and context-aware systems. Lemmens et al. (2020) adopted an ensemble approach that integrated four disparate models: LSTM, CNN, MLP, and SVM. Each model was tailored to capture different textual features like emojis, hashtags, and word embeddings, resulting in an F1-score of 74%. Khotijah et al. (2020) employed LSTM but took it a step further by incorporating Paragraph2vec for context extraction. Their study was unique in its finding that reversing the word order in tweets could lead to better results, achieving an F1-score of 87.03% on their Indonesian Twitter dataset. Handoyo (2021) focused on the challenge of unbalanced datasets, employing the RoBERTa architecture (2019) coupled with GloVe for data augmentation. This methodology improved the model's ability to classify non-sarcastic text, contributing to an overall F1-score enhancement.

The field of sarcasm detection has grown considerably, with an increasing number of sophisticated methodologies. While traditional machine learning approaches continue to be effective, especially with advanced feature engineering, the trend is clearly moving toward more intricate deep learning and ensemble methods. These methods often offer superior performance metrics and are becoming increasingly nuanced in their ability to capture context and other complex linguistic features inherent in sarcasm.

2.2. IndoGPT

IndoGPT (Cahyawijaya, et al., 2021) is a decoder-only model based on GPT-2 (Radford, et al., 2019). It consists of 12 modified transformer decoder layers, where layer normalization was placed before each sub-block. Furthermore, an extra layer normalization step was incorporated subsequent to the last self-attention block. Each decoder has 12 attention heads, an embedding dimensionality of 768, and a feed-forward network size of 3072. This model weighs in at approximately 117 million parameters and has the maximum sequence length of 1024 tokens. It was also pre-trained on three languages, Indonesian, Sundanese, and Javanese, utilizing autoregressive language modeling objective.

2.3. IndoBERT & IndoBERTweet

IndoBERT (Koto, Rahimi, Lau, & Baldwin, 2020) is a pre-trained model that utilizes BERT architecture (Devlin, Chang, Lee, & Toutanova, 2019), an encoder-only part of Transformer architecture. The model comprises 12 hidden layers with dimensions of 768, 12 attention heads, and feed-forward hidden layers featuring a dimensionality of 3072. While BERT was trained using both Masked Language Model (MLM) and Next Sentence Prediction (NSP), IndoBERT only utilized MLM for its pre-training. The model was trained using over 220 million words collected from various sources, including Indonesian Wikipedia, news articles, and the Indonesian Web Corpus.

Extended from IndoBERT, IndoBERTweet (Koto, Lau, & Baldwin, 2021) is a domain-specific pre-trained model for social media data, with the primary purpose of replacing the entire IndoBERT's vocabulary with a vast Twitter dataset consisting of 26 million Indonesian tweets with 409 million words. Unlike IndoBERT's maximum token sequence length of 512, IndoBERTweet has a maximum length of 128 tokens. Additionally, the model underwent a simple average of subword embeddings, which significantly reduced the domain-adaptive pretraining overhead by 80%. To assess its effectiveness, it underwent evaluation across seven datasets, focusing on four downstream tasks; named entity recognition (NER), hate speech detection, sentiment analysis, and emotion classification.

3. Methodology

As depicted in Fig. 1, Our objective is to identify whether a particular collection of tweets contains sarcasm. We evaluated the performance of several deep learning models for classifying sarcasm in

Indonesian tweets. Specifically, we compared the performance of three different types of transformer-based models, IndoGPT, IndoBERT, and IndoBERTweet, as well as LSTM-based model proposed by Khotijah et al. (2020). Tweet data in the form of text is collected and stored as a dataset. Then, each tweet is going through preprocessing to obtain clean datasets which were partitioned for training, validation, and testing. Following preprocessing, hyperparameter optimization is carried out for each model and trained on the training dataset, resulting in models that are fine-tuned. These fine-tuned models were subsequently evaluated using the testing dataset.

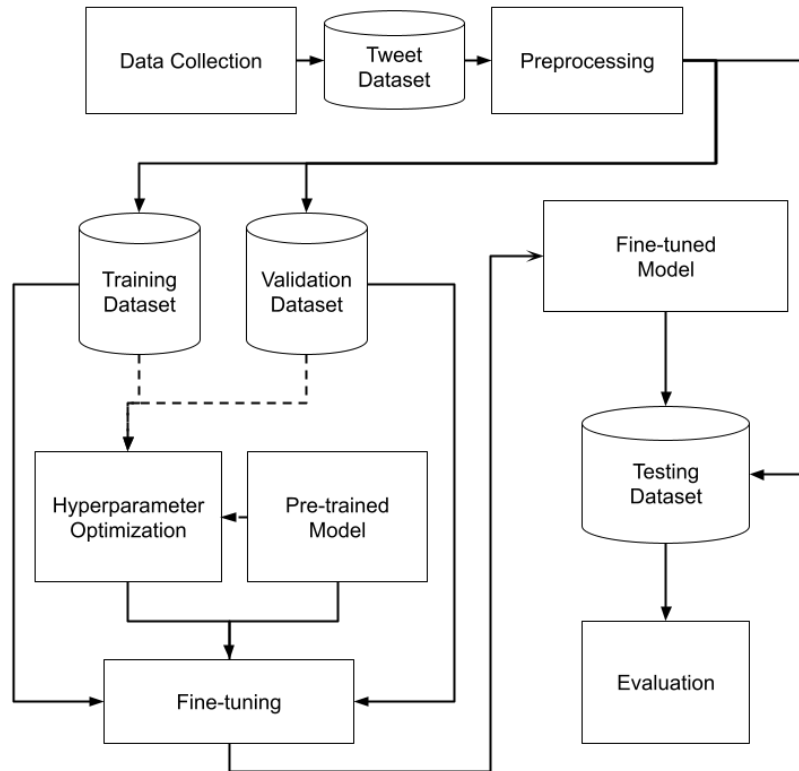


Fig. 1: Process Diagram

3.1. Data Collection

The dataset was sourced from Khotijah et al. (2020) and encompasses a collection of 8700 tweets. These tweets were collected using the Twitter API during the period between March 2013 and February 2020. The dataset is categorized into two distinct categories: sarcastic and non-sarcastic tweets, each containing 4350 entries. Additionally, a separate set of 120 tweets was curated from the same source for testing purposes. It consists of 56 sarcastic and 64 non-sarcastic tweets.

Annotation was carried out based on criteria for positive and negative sentiments. This process performed manually by five individuals, who were unfamiliar with the context of the tweets or the identities of the tweet authors. Any ambiguities in labeling were resolved by consulting a linguistics expert. For the purpose of data validation, another group of five individuals was involved. In instances where disagreements occurred among the validator, a consensus was achieved through a majority voting system.

3.2. Preprocessing and Tokenization

In the process of dataset preprocessing, a multi-step method was conducted as follows:

1. Removing noisy characters resulted from emojis parsed with UTF-8 encoding during data collection (e.g. $\tilde{A}f\hat{A}^{\circ}\zeta\grave{a}\hat{c}\grave{z}-\hat{A}_j\text{☹}“\dots\grave{s}\grave{E}\grave{\alpha},\dagger^{2TM}1/4”\text{©}|\grave{Y}^{\sim}1-\grave{S}^{\wedge}\grave{Z}\acute{I}”\text{€r}^{\text{oa}}\text{®}\grave{y}$)

2. Eliminating unicode characters as they can increase data complexity. These characters are represented as escape sequences in the text (e.g., \u2026).
3. Case folding. Converting all text to lowercase helps to ensure consistency and reduce the number of unique words and phrases.
4. Replacing mentions (e.g., "@tvOneNews") with "@USER" and URLs (e.g., "https://t.co/2G2azHJx3g") with "HTTPURL". This step is undertaken to account for the fact that sarcasm in tweets may function as a response to an individual or as a reaction to online articles or news stories.
5. Normalizing slang words based on dictionary by Ibrohim & Budi (2019). This refers to the process of converting informal language or words that are commonly used in casual conversation, but not necessarily in formal writing, into more standard language.
6. Removing stop words. Removing common words that do not carry much meaning can simplify and reduce data complexity. This step is done using NLTK's Indonesian stop words library.

As part of natural language processing, tokenization involves segmenting text into smaller units, commonly called tokens. These tokens are required to be mapped to a numerical representation in the form of vectors called embedding. Khotijah et al. (2020) adopt Keras Tokenizer for their LSTM model and IndoGPT utilizes SentencePiece in conjunction with the byte-pair encoding (BPE) algorithm. Both IndoBERT and IndoBERTtweet employ WordPiece algorithm introduced by Wu et al. (2016). To demonstrate the difference in tokenization, consider the sentence "*proses digitalisasi dapat dinormalisasi agar proses administrasi berjalan dengan optimum*" (translated: "the digitalization process can be normalized so that administrative processes run optimally"). The differences between the tokenization results are shown in Table 1.

Table 1: Tokenization Comparison

Tokenization Method	Tokenization Result
Keras Tokenizer	['proses', 'digitalisasi', 'dapat', 'dinormalisasi', 'agar', 'proses', 'administrasi', 'berjalan', 'dengan', 'optimum']
SentencePiece + BPE (IndoGPT's corpus)	['_proses', '_digit', 'alisasi', '_dapat', '_din', 'orm', 'alisasi', '_agar', '_proses', '_administrasi', '_berjalan', '_dengan', '_optimum']
WordPiece (IndoBERT's corpus)	['proses', 'digital', '###isasi', 'dapat', 'dino', '###', '###mal', '###isasi', 'agar', 'proses', 'administrasi', 'berjalan', 'dengan', 'optim', '###um']
WordPiece (IndoBERTtweet's corpus)	['proses', 'digitalisasi', 'dapat', 'dino', '###rm', '###alisasi', 'agar', 'proses', 'administrasi', 'berjalan', 'dengan', 'optim', '###um']

3.3. Experimental Setup

To find the optimal hyperparameters for each model, we fine-tuned IndoGPT, IndoBERT, and IndoBERTtweet using Optuna framework (2019). The following steps are conducted:

1. Search space was defined. In Table 2, lower endpoint and upper endpoint for each hyperparameter that affects model training are define. The learning rate determines the size of the step to be taken in each iteration towards the minimum of a loss function. Weight decay is utilized as a regularization method to hinder overfitting. The number of epochs is the number of times that the entire training dataset is shown to the model during training. Both training and evaluation batch size specify how many input sequences, such as sentences or documents, are processed in a single forward and backward pass through the model.

Table 2: Search Space Configuration

Parameter	Lower Endpoint	Upper Endpoint
Learning Rate	4×10^{-5}	6×10^{-5}

Weight Decay	4×10^{-5}	6×10^{-5}
Epoch	2	5
Training Batch Size	8	16
Evaluation Batch Size	8	16

- Objective was defined by creating a function that returns the model's loss value during trials on training and validation dataset.
- A study object was created to manage hyperparameter search and optimization process. It was configured to minimize the loss of the model.
- Optimization process with $n = 3$ trials was executed, where each trial represents an evaluation of the objective function with a specific set of hyperparameters. We used Tree-structured Parzen Estimator (TPE) algorithm as the sampling strategy. As articulated on Equation (1), TPE defines two probability density functions, $l(x)$ and $g(x)$, which represent the probability density of x being associated with a "low" performance or a "good" performance, respectively, relative to the best performance achieved so far (y^*). In conjunction with Asynchronous Successive Halving Algorithm (ASHA), it can focus the search on hyperparameters that have a higher probability of leading to good performance and prune unpromising trials early.

$$p(x|y) = \begin{cases} l(x) & \text{if } y < y^* \\ g(x) & \text{if } y \geq y^* \end{cases} \quad (1)$$

- Results were evaluated from the study object after all the trials from optimization are completed. It yields the hyperparameters that result in the lowest training loss observed throughout the trials.

The dataset that contains 8700 tweets is partitioned at a ratio of 9:1. Specifically, the training dataset includes 7830 tweets, while the validation dataset consists of 870 tweets. Stratified sampling was also employed to ensure an equitable distribution of both sarcastic and non-sarcastic tweets across these datasets. For the evaluation phase, the testing dataset was utilized.

4. Results and Discussion

A range of hyperparameters adjustment were automated for every pre-trained model to identify configurations that would minimize the loss function. We leveraged Tree-structured Parzen Estimator for hyperparameter sampling and the Asynchronous Successive Halving Algorithm for early termination of less promising trials. Utilizing these techniques not only ensures a more efficient search for optimal hyperparameters but also optimizes the computational resources and time invested in the training process. The comparative results are encapsulated in Table 3.

Table 3: Hyperparameter Optimization Result

Model	Learning Rate	Weight Decay	Epoch	Training Batch Size	Evaluation Batch Size
IndoGPT	4.07×10^{-5}	4.57×10^{-5}	2	16	16
IndoBERT	4.44×10^{-5}	5.72×10^{-5}	3	16	16
IndoBERTweet	5.32×10^{-5}	5.97×10^{-5}	2	8	8

The varying performance levels across different models are assessed using a Confusion Matrix. This matrix provides an overview of the prediction outcomes for a classification issue. Its purpose is to evaluate the dependability of the suggested approach and the effectiveness of the method in detecting sarcasm. The formula for the confusion matrix is described on Equation (2), Equation (3), Equation (4), and Equation (5), where TP is true positive, TN is true negative, FP is false positive, and FN is false negative.

$$\text{Accuracy} = \frac{TP}{TN + FP + FN + TP} \quad (2)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4)$$

$$\text{F1 Score} = \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

The NLP model performances on validation and testing dataset showcase the impact of various factors on model performance, including tokenization algorithms, pretraining strategies, and domain-specific adaptations. The result is displayed on Table 4 and Table 5.

Table 4: Model Performance on Validation Dataset

Model	Accuracy	Precision	Recall	F1-score
LSTM (Khotijah et al.)	95.40%	95.63%	95.19%	95.40%
IndoGPT (fine-tuned)	95.76%	97.61%	93.84%	95.68%
IndoBERT (fine-tuned)	94.70%	97.25%	92.02%	94.55%
IndoBERTweet (fine-tuned)	95.09%	98.36%	91.72%	94.92%

Table 5: Model Performance on Testing Dataset

Model	Accuracy	Precision	Recall	F1-score
LSTM (Khotijah et al.)	88.33%	83.92%	90.38%	87.03%
IndoGPT (fine-tuned)	87.83%	80.30%	98.21%	88.32%
IndoBERT (fine-tuned)	88.08%	81.35%	96.79%	88.36%
IndoBERTweet (fine-tuned)	89.00%	82.84%	96.43%	89.11%

IndoGPT demonstrates strongest performance in the validation dataset, with an accuracy of 95.76% and an F1 score of 95.68%. This can be attributed to its pretraining on three related languages, Indonesian, Sundanese, and Javanese, enabling it to capture essential linguistic features when fine-tuned on the datasets. In addition, SentencePiece tokenization with byte-pair encoding algorithm can also contribute to its performance. However, the model accuracy and F1 scores were slightly lower in the testing dataset, indicating reduced generalizability across completely different datasets. It also achieved the lowest recall of 80.30% in testing dataset, indicating that its ability to understand the context is reduced due to masked multi-head attention on the architecture that only enables each position to attend all preceding positions up to the current position by subtracting succeeding positions with ∞ (infinity).

Both IndoBERT and IndoBERTweet show almost the same performance in terms of accuracy and F1 score on validation and testing datasets. Despite using the same WordPiece tokenization method, IndoBERTweet's vocabulary that entirely replaced IndoBERT's corpus with the Twitter dataset results in slightly better performance both in the validation and testing dataset. It indicates that domain-specific pre-training increases IndoBERTweet's performance on Twitter data. The model also surpasses LSTM, used here as the baseline, on the testing dataset. Unlike LSTM, which processes embeddings from scratch, IndoBERTweet already has a foundational understanding of the language acquired during its pre-training phase. Additionally, while LSTM processes representations in a sequential and unidirectional manner, IndoBERTweet processes them concurrently and bidirectionally, leveraging the self-attention mechanism.

The dataset used for training the models consists of 8700 tweets. It is worth noting that this dataset might not be wholly representative of the broader Twitter user base in Indonesia. The limited dataset size may introduce bias and impact the generalizability of the findings. Furthermore, the dataset's relatively small scale could make the model susceptible to overfitting, resulting in high variance when applied to new, unseen data.

From a linguistic perspective, Indonesian language is rich in idiomatic expressions, slang, and dialectal variations. This poses challenges in creating a model that performs universally well. Furthermore, the problem of understanding sarcasm often requires preceding contextual information that the text responds to, which is sometimes lost or inadequately captured in the model. These linguistic complexities could affect its accuracy and robustness in classifying sarcasm correctly.

As for the model design, IndoBERTweet is based on the Transformer architecture and follows specific hyperparameter configurations. While these hyperparameters were chosen to optimize performance based on preliminary experiments, different architectural could potentially yield different outcomes. For instance, increasing the number of attention heads might enhance the model's ability to capture more nuanced relationships in the data, albeit at a higher computational cost. Additionally, the model was developed with a maximum token sequence length of 128, which might be insufficient for capturing longer sarcastic statements, especially considering the 280-character limit of recent tweets.

5. Conclusion

This study investigates the critical roles of tokenization algorithms, pre-training methodologies, and domain-specific vocabulary adaptations in influencing the performance of Natural Language Processing (NLP) models for the classification of sarcastic tweets in Indonesian. These constituent factors merit careful consideration in future model selection or development endeavors, as they are pivotal in encapsulating the linguistic complexities and ensuring robust performance across diverse datasets.

In terms of empirical results, the fine-tuned IndoBERTweet model achieved an accuracy of 89.00%, thereby outperformed models such as IndoGPT, IndoBERT, and LSTM, affirming its position as the state-of-the-art. It is thereby concluded that IndoBERTweet presents a promising approach for sarcasm classification within the corpus of Indonesian-language Twitter data. This improvement has potential applications in multiple domains, including sentiment analysis, customer reviews, political insights, and social media monitoring.

Further research could focus on enhancing the model's potential in several ways. For instance, refining tweet-scraping methodologies to include emoji processing could offer additional contextual support. Additionally, the possibility of improving context representation could be investigated by using IndoBERTweet embeddings for feature extraction. These embeddings could then be combined with other word embeddings, like Paragraph2Vec.

References

- Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, (pp. 2623–2631).
- Ashok, D. M., Ghanshyam, A. N., Salim, S. S., Mazahir, D. B., & Thakare, B. S. (2020, June). Sarcasm Detection using Genetic Optimization on LSTM with CNN. *2020 International Conference for Emerging Technology (INCET)*. IEEE. doi:10.1109/incet49848.2020.9154090
- Bouazizi, M., & Ohtsuki, T. (2017). A Pattern-Based Approach for Multi-Class Sentiment Analysis in Twitter. *IEEE Access*, 5, 20617-20639. doi:10.1109/ACCESS.2017.2740982

- Cahyawijaya, S., Winata, G. I., Wilie, B., Vincentio, K., Li, X., Kuncoro, A., . . . Fung, P. (2021). IndoNLG: Benchmark and Resources for Evaluating Indonesian Natural Language Generation. doi:10.48550/ARXIV.2104.08200
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, June). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4171–4186). Minneapolis: Association for Computational Linguistics. doi:10.18653/v1/N19-1423
- Dixon, S. J. (2022, November). Countries with most Twitter users 2022. *Countries with most Twitter users 2022*. Statista. Retrieved from <https://www.statista.com/statistics/242606/number-of-active-twitter-users-in-selected-countries/>
- Dutta, S., & Mehta, A. (2021). Unfolding Sarcasm in Twitter Using C-RNN Approach. *Bulletin of Computer Science and Electrical Engineering*, 2, 1–8.
- Eke, C. I., Norman, A. A., & Shuib, L. (2021, June). Multi-feature fusion framework for sarcasm identification on twitter data: A machine learning based approach. (J. Rashid, Ed.) *PLOS ONE*, 16, e0252918. doi:10.1371/journal.pone.0252918
- Handoyo, A. T., Hidayaturrehman, & Suhartono, D. (2021). Sarcasm Detection in Twitter – Performance Impact while using Data Augmentation: Word Embeddings. *Sarcasm Detection in Twitter – Performance Impact while using Data Augmentation: Word Embeddings*. arXiv. doi:10.48550/ARXIV.2108.09924
- Ibrohim, M. O., & Budi, I. (2019). Multi-label Hate Speech and Abusive Language Detection in Indonesian Twitter. *Proceedings of the Third Workshop on Abusive Language Online*. Association for Computational Linguistics. doi:10.18653/v1/w19-3506
- Khotijah, S., Tirtawangsa, J., & Suryani, A. A. (2020). Using LSTM for Context Based Approach of Sarcasm Detection in Twitter. *Proceedings of the 11th International Conference on Advances in Information Technology*. New York, NY, USA: Association for Computing Machinery. doi:10.1145/3406601.3406624
- Koto, F., Lau, J. H., & Baldwin, T. (2021). IndoBERTweet: A Pretrained Language Model for Indonesian Twitter with Effective Domain-Specific Vocabulary Initialization. *IndoBERTweet: A Pretrained Language Model for Indonesian Twitter with Effective Domain-Specific Vocabulary Initialization*. arXiv. doi:10.48550/ARXIV.2109.04607
- Koto, F., Rahimi, A., Lau, J. H., & Baldwin, T. (2020). IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP. *Proceedings of the 28th COLING*.
- Lemmens, J., Burtenshaw, B., Lotfi, E., Markov, I., & Daelemans, W. (2020). Sarcasm Detection Using an Ensemble Approach. *Proceedings of the Second Workshop on Figurative Language Processing*. Association for Computational Linguistics. doi:10.18653/v1/2020.figlang-1.36
- Lieberman, A., & Schroeder, J. (2020, February). Two social lives: How differences between online and offline interaction influence social outcomes. *Current Opinion in Psychology*, 31, 16–21. doi:10.1016/j.copsyc.2019.06.022
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., . . . Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. arXiv. doi:10.48550/ARXIV.1907.11692
- Misra, R. (2019, July). News headlines dataset for sarcasm detection. *News headlines dataset for sarcasm detection*. Retrieved from <https://www.kaggle.com/datasets/rmisra/news-headlines-dataset-for-sarcasm-detection>

Oprea, S., & Magdy, W. (2019). iSarcasm: A Dataset of Intended Sarcasm. *iSarcasm: A Dataset of Intended Sarcasm*. arXiv. doi:10.48550/ARXIV.1911.03123

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., & others. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1, 9.

Rahayu, D. A., Kuntur, S., & Hayatin, N. (2018, October). Sarcasm Detection on Indonesian Twitter Feeds. *2018 5th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*. IEEE. doi:10.1109/eecsi.2018.8752913

Ren, L., Xu, B., Lin, H., Liu, X., & Yang, L. (2020, August). Sarcasm Detection with Sentiment Semantics Enhanced Multi-level Memory Network. *Neurocomputing*, 401, 320–326. doi:10.1016/j.neucom.2020.03.081

Yunitasari, Y., Musdholifah, A., & Sari, A. K. (2019, January). Sarcasm Detection For Sentiment Analysis in Indonesian Tweets. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 13, 53. doi:10.22146/ijccs.41136