JOURNAL OF CULTURAL ANALYSIS AND SOCIAL CHANGE

Journal of Cultural Analysis and Social Change,
2025, 10(3), 918-926
ISSN: 2589-1316

Sarcasm Detection in Arabic Text using Artificial Intelligence-Based Modeling

Soleman Alzobidy[1*]

[1] *Department of English Language and Translation Studies, College of Sciences and Theoretical Studies, Saudi Electronic University, Saudi Arabia, s.alzobidy@seu.edu.sa, https://orcid.org/0009-0008-6506-305X*

***Corresponding Author:** s.alzobidy@seu.edu.sa

Citation: Alzobidy, S. (2025). Sarcasm Detection in Arabic Text using Artificial Intelligence-Based Modeling, *Journal of Cultural Analysis and Social Change*, 10(3), 918-926. https://doi.org/10.64753/jcasc.v10i3.2529

Published: November 28, 2025

ABSTRACT

Social media's pervasiveness in our daily lives has made it possible for people to voice their thoughts without facing consequences. As a result, comments made online about anything or anyone frequently include sarcastic overtones. To better understand attitudes and social expressions in digital communication, researchers are growing more and more interested in sarcasm detection. However, because sarcastic remarks are subjective, context-dependent, and culturally nuanced, identifying them can be challenging. This task becomes even more difficult for Arabic, a language that is morphologically rich and diverse, with numerous dialects. Consequently, an effective sarcasm detection model for Arabic must be developed. This paper proposes an AraBERT transformer-based model for sarcasm detection in Arabic text. A dataset of approximately 10,500 human-annotated Arabic tweets was collected from publicly available digital media platforms. The dataset has a skewed distribution of sarcastic and non-sarcastic statements, and various strategies were applied to address this imbalance during training. The model was trained and tested on unseen datasets, achieving an accuracy of 92.46% and specificity of 95.82%. Its performance was also measured using precision, recall, F1-score, specificity, and accuracy. For Arabic sarcasm detection, the suggested model performs better than a number of alternative models already in use. The suggested model performs admirably when it comes to identifying irony in Arabic text.

Keywords: AraBERT, Arabic NLP, Sarcasm Detection, Social Media Analysis, Digital Communication.

INTRODUCTION

Social media platforms provide users with a place to express their opinions freely. Their opinions might carry the tone of sarcasm, in which case understanding it is both a cognitive and practical challenge. (Pang & Lee, 2008; Hutto et al., 2013). Sarcasm, a form of verbal irony, often conveys a meaning opposite to the literal interpretation of the words used (Joshi et al., 2017). Sarcastic remarks may serve to amuse, mock, or subtly critique, with their meaning frequently dependent on context, tone, or nonverbal cues. In other words, sarcasm language carries a meaning that is the opposite of the literal meaning, which makes it a problem for the text classifier systems. This environment encourages open expression but also underscores the need to comprehend diverse viewpoints and social cues in online communication (Bamman et al., 2014). As sarcasm relies mainly on nonverbal cues, suggestion, and context, it presents a particular challenge to direct methods of detection to identify or extract them. Hence, from this argument, the need for this research is raised due to its impact on public opinion assessment, customer service, policy making, and education and media analyses.

The detection of sarcasm has thus attracted significant attention from researchers, who aim to interpret user comments on social networking sites accurately. Such efforts can help organizations assess customer satisfaction, shape policy decisions, and analyze sentiments in fields such as education (Alhazmi et al., 2023). This involves using natural language processing (NLP) and text mining to glean sentiment from text obtained through online platforms (Goswami et al., 2025). Based on the interpretation of opinions in digital communication and the categorization of such opinions as either positive, negative, or neutral, sentiment analysis is a subfield of NLP.

However, this analysis remains limited when dealing with texts that involve paradox or irony. The task of the AI model, therefore, becomes one of navigating this hyperreal communication to decode the underlying reality it obscures. Sarcasm detection is distinct from general sentiment analysis due to the former's nuanced and often contradictory nature (Alqahtani et al., 2023). Accurately identifying sarcasm is therefore essential for understanding the true sentiment behind online interactions. Automatic sarcasm detection employs computational methods to predict sarcasm, and ongoing research seeks to enhance the accuracy of these models.

Nonetheless, the majority of these initiatives focus on the English language (Talafha et al., 2021; Davidov et al., 2010). Sarcasm detection, like other NLP tasks, often employs deep learning techniques. The inherent ambiguity and subjectivity of sarcasm make its detection particularly challenging, as interpretations can vary based on individual, cultural, and contextual factors. These challenges are compounded for languages like Arabic, which is spoken by approximately 447 million people worldwide and is the primary language in 22 Arab countries (Muaad et al., 2022). As the fifth most spoken language globally, Arabic possesses rich morphological features and numerous dialects, adding layers of complexity to NLP tasks such as sentiment analysis and sarcasm detection (Habash, 2010). A systematic review of Arabic-to-English machine translation has identified key linguistic challenges, including word sense disambiguation, rich morphology, and handling named entities, which are also central to the accurate detection of sarcasm (Almayyah & Alzobidy, 2023). Researchers have endeavored to address dialectal and semantic variations in Arabic NLP (Bouamor et al., 2018). Arabic's spelling and morphological complexity render it comparatively under-resourced and linguistically challenging relative to English. As a result, research on sarcasm detection in Arabic remains limited (Elgabry et al., 2021). Nonetheless, the potential for developing NLP applications for Arabic—including sentiment analysis, sarcasm detection, and part-of-speech tagging—is substantial. The interpretation of sarcastic statements varies among individuals and often relies on unconventional or opposite uses of words, making AI-based detection particularly demanding and necessitating robust models.

Recently, transformer-based models have gained prominence in Arabic sarcasm detection (Vaswani et al., 2017). BERT-based models, especially those adapted for Arabic, have seen widespread adoption (Rahma et al., 2023). Bidirectional Encoder Representations from Transformers (BERT) have proven effective across various NLP tasks (Devlin et al., 2019). While BERT has been customized for Arabic, its applications remain somewhat constrained (Alammary, 2022; Safaya et al., 2020). This study employs the AraBERT model (Antoun et al., 2020), a BERT-based architecture tailored for Arabic. The performance of such models depends not only on their underlying architecture but also on factors such as corpus size and quality, feature extraction methods, and parameter configuration.

This study is organized into five sections: Section two presents a review of the literature on automatic sarcasm detection in Arabic. Section three outlines the architecture of the proposed system, detailing data preparation and implementation. Section four presents result and compares the proposed model to existing Arabic sarcasm detection approaches using metrics including accuracy, specificity, recall, precision, and F1-score. Section 5 concludes the study and suggests directions for future research.

LITERATURE REVIEW

In general, sarcasm detection is a challenging task for NLP. It's a sophisticated kind of communication. Sentiment detection may be aided by applying a straightforward interpretation of the text. However, basic sentiment analysis methods are typically insufficient for sarcasm identification. Much progress has been made in this area over the past ten years. Nonetheless, the majority of these studies have focused on the English language. There is not much work done in other languages. The absence of adequate resources is one of the leading causes. Arabic still lacks the resources necessary to carry out NLP tasks like sarcasm detection, despite being one of the most spoken languages. Nonetheless, a number of efforts are underway to create a sarcasm detection model that works well in the intricate Arabic language. Numerous approaches have been tried by researchers, including rule-based strategies, hybrid systems, neural network-based structures, and different machine-learning methodologies. A few of these methods are reviewed in brief in this section.

The earliest approaches were built on traditional machine learning. Studies like that of Al-Ghadhban et al. (2017) applied classifiers like Naïve Bayes to the problem, providing a simple but important starting point. While these methods laid the groundwork, they often struggled because sarcasm is so dependent on unspoken context and the author's intent—things that are difficult to capture with rigid rules or basic statistics.

Recognizing these limitations, the field began to embrace more powerful deep learning models. For example, Farha and Magdy (2020) used a Bidirectional LSTM (BiLSTM) network, which was better at understanding the context of words in a sentence. Their work suggested that the key to better detection might lie in incorporating even deeper cultural and world knowledge, pointing toward the need for more sophisticated models. Around the same time, researchers began to focus just as much on the data itself as on the models. Israeli et al. (2021) showed that techniques like data augmentation and down-sampling could significantly boost a model's performance by creating more robust and balanced training data.

The real breakthrough, however, has come with the arrival of transformer-based models like BERT. These models, pre-trained on massive amounts of text, revolutionized NLP by developing a deep, contextual understanding of language. Researchers were quick to apply them to Arabic. Muaad et al. (2022) championed AraBERT, a version of BERT tailored for Arabic, as a top-tier tool for sarcasm detection. Soon after, Israeli et al. (2021) demonstrated that combining AraBERT with smart data augmentation techniques could yield impressive results, directly tackling the problem of scarce data. Similarly, Elgabry et al. (2021) used AraBERT's power to grapple with the immense diversity of Arabic dialects. "The rich and complex morphology of Arabic, where a single word can represent an entire English sentence, presents a significant hurdle for computational models (Almaaytah & Alzobidy, 2023; Habash, 2010)."

Of course, even the best models are useless without good data. The creation of the first datasets, like the human-annotated tweets provided by Talafha et al. (2021), was a vital step forward. However, this work also highlighted a recurring problem: these datasets are often small and imbalanced, with far more examples of one class (usually sarcasm) than the other. This imbalance can severely hamper a model's ability to learn accurately and generalize well.

In summary, the quest to detect sarcasm in Arabic text has evolved from simple rule-based systems to powerful deep-learning models. The advent of AraBERT has particularly marked a turning point. However, the research remains fragmented. Progress is still constrained by the lack of a large, balanced, and standardized benchmark dataset that the entire research community can use. This study seeks to build upon this existing foundation by constructing a substantial new dataset of Arabic tweets, rigorously addressing the class imbalance issue, and thoroughly evaluating a fine-tuned AraBERT model to contribute a reliable benchmark for future work in Arabic sarcasm detection.

Proposed Model

System Architecture

The system architecture of the suggested sarcasm detection model is depicted in Figure 1. Preprocessing, training, and testing are the three main elements of the suggested methodology. The preprocessing module handles data collection, synthesis, cleaning, and transformation. The information on Arabic writings that contain sarcasm was gathered from a variety of publicly accessible sources. The data comes in a variety of formats and includes a variety of superfluous tags, hyperlinks, emoticons, and other elements, for the training module to comprehend and use it in the training process, it must be cleansed and changed into a structure that works. The training module receives the data when it is prepared.

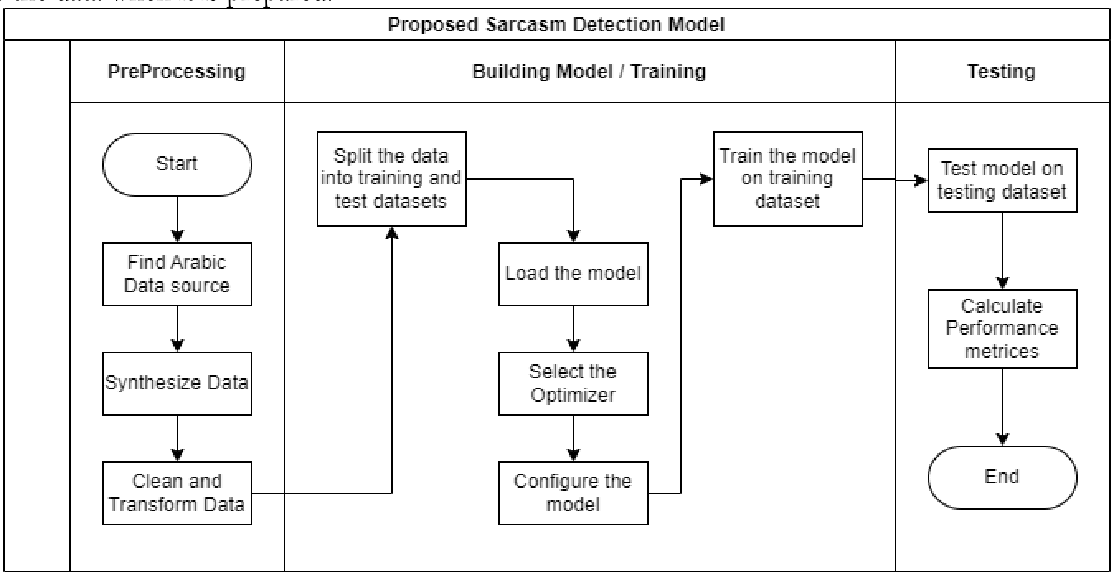

Figure 1: Proposed Architecture for Arabic Sarcasm Detection Model

The neural network's training configuration is established after loading the transformer model AraBERT into the memory. The model has been optimized using the AdamW optimizer (Loshchilov & Hutter, 2017). AdamW is an optimization method that fine-tunes the models using weight decay. After the model's configuration has been established. It undergoes five epochs of training on the training dataset. The system moves into the testing phase following model training. The held-out data is used to test the trained model in the testing module. The performance metrics for evaluating the effectiveness of the suggested model are computed using the anticipated and actual data points.

Dataset

Data on Arabic texts that contain sarcasm have been compiled for this research project from a variety of publicly accessible sources. About 10,500 tweets in Arabic are included in the collection. It has about 8800 tweets that are devoid of sarcasm. The sarcastic tweets date back to approximately 1700. The total number of tweets for each class is shown in Figure 2. In order to train the model to classify the input into binary categories, the dataset has been prepared. Figure 2's dataset illustration makes it clear that the dataset is skewed. Consequently, appropriate steps should be taken to prevent prediction bias in favour of the majority class. The training procedure explains the specifics of the actions taken to address the dataset's skewness in the subsection that follows.

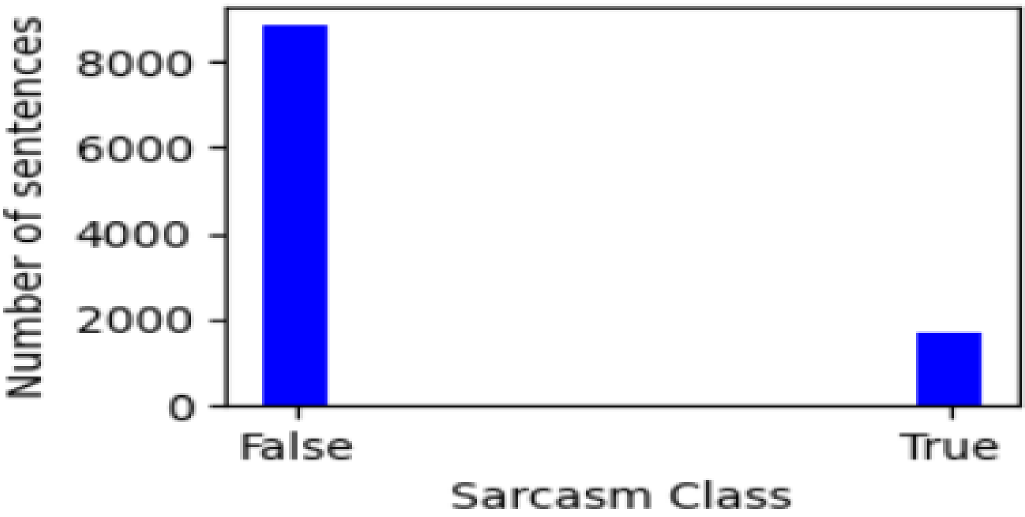

Figure 2: Number of Sarcastic and Non-Sarcastic Data Points

One of the most important tasks in machine learning modelling is text preparation, which is necessary for the training process to enable learning adequately. The preprocessing step converts the data into a format that neural network design can use for learning. The data must be pre-processed to ensure consistency in its structure and representation because it has been synthesized from several sources. Additionally, the data has been preprocessed to eliminate extraneous characters, tags, emoticons, HTML markups, URLs, and other undesirable information. Additionally, it makes an effort to eliminate unnecessary punctuation, stop words, and other components from the raw data. The presence of undesirable words or characters could result in ineffective learning. "Tashkeel" and "tatweel" have also been eliminated from Arabic text. The diacritical symbols used to denote short vowels and other phonetic qualities are referred to as tashkeel. Some Arabic letters have a horizontal line called tatweel appended to them to denote lengthy vowels or consonants that are not pronounced with tashkeel. Training and test data have been separated from the final preprocessed data. Twenty percent of the processed data's tweets were chosen at random to serve as test data. Here are some examples of preprocessing before and after, as seen in Table 1.

Table 1: Data Samples before and after preprocessing

Raw Text	Preprocessed
"نصيحه ما عمرك اتنزل لعبة سوبر ماريو مش زي ما ...	"نصيح ه ما عمر ك اتنزل لعب +ة سوبر ماريو مش ...
"نادين_نسيب_نجيم ❤️❤️❤️مجلة #ماري_كلير 💭 #ملكة#...	"نادين _ نسيب _ نجيم مجل +ة #ماري _ كلير # ...
"@Alito_NBA اتوقع انه بيستمر"	"[بريد] أتوقع ان ه ب يستمر
"@KSA24 يعني "بموافقتنا "لأن دمشق صايرة موسكو"	[بريد] يعني "ب موافق ت نا "ل أن دمشق صاير ...
"RT @alaahmad20: قائد في الحرس يعترف بفقدان ال...	"RT [مستخدم] : [قائد في ال حرس يعترف ب فقدان ا ...

```
Set the model to the training mode
Initialize the number of epochs
For each epoch repeat:
Iterate over batches in the training data loader
Get the batch (input_ids, attention_mask, labels)
Zero the gradients in the optimizer

Forward Pass:
Perform a forward pass with autocasting
Get the model outputs, including the loss
Backward Pass:
Scale the loss with the gradient scaler
Perform a backward pass

Step the optimizer to update model parameters
Update the gradient scaler

Learning Rate Scheduler:
Adjust the learning rate based on the scheduler
```

Figure 3: Training Process

Implementation

The proposed model has used the AraBERTv02 model (Antoun et al., 2020)for developing the sarcasm detection model. This model has been developed based on the BERT model (Devlin et al., 2019). The pre-trained AraBERT model was fine-tuned on our labeled dataset for this research study. The architecture of the AraBERT has been adopted from the HuggingFace library. However, the model has been configured to handle the skewness present in the data during the training process. To address the imbalanced class distribution, class weights were calculated inversely proportional to class frequencies and integrated into the Cross-Entropy loss function during training. The experiments have been conducted on the Google Colab Pro platform. The Python programming language has been used for conducting the experiment. The supporting libraries, such as scikit learn, PyTorch Transformers, pandas, matplotlib, etc., have been used as per the requirement. The model has been fine-tuned on the dataset that has been synthesized for this research study. Once the dataset has been finalized, the model parameters have been tuned for training the model. AraBERT model uses the BERT-base configuration. It (Vaswani et al., 2017) comprises 12 encoder blocks and 12 attention blocks.

Heads, a maximum sequence length of 512, 768 hidden dimensions, and approximately 110 million parameters. The model (Vaswani et al., 2017) is considered to be the foundation of achieving the best results in different NLP tasks across multiple natural languages. However, the compatibility with the Arabic language is not that great. To enhance this compatibility, AraBERT incorporates additional preprocessing steps before training the model. The AraBERT model implements Masked Language Modeling, incorporating whole-word masking. Masking the whole word improves the pre-training task. BERT model training comprises two steps, which are masked language model and next sentence prediction. AraBERT is trained using approximately 70 million sentences from various sources (Zhu et al., 2015; El-Khair, 2016; Zeroual et al., 2019). In this research study, the model has been trained with our labeled training data so that it learns to predict the correct class for each sequence. The training process follows the following steps:

Following a number of tests, five training epochs are used. Because there are different numbers of representatives of each class (sarcastic tweets and non-sarcastic tweets). According to Muaad et al. (2022), it is among the main obstacles in Arabic NLP. In order to address the imbalanced class difficulties, various weights have been assigned during the training procedure. A bigger batch size was used during the training procedure in order to lower the model's variance. The minority class performs better when a higher batch size is used. However, a trial-and-error approach has been used to evaluate various batch sizes in order to train the model with the best speed and variance reduction.

By raising the batch size, this study also aims to take advantage of GPU acceleration and parallel processing. However, given the available resources, a batch size of 128 was chosen because the larger batch sizes do not fit into GPU memory. A learning rate scheduler known as warm-up has been utilized to modify the learning rate throughout the training process. During the training process, the 5-fold cross-validation procedure was employed.

DISCUSSION AND RESULTS

The findings of our proposed model are shown in this part. A held-out unseen test set that was held back for testing and comprised twenty percent of the total data collected was used to validate the model. A number of evaluation metrics, including accuracy, precision, recall, F1-score, and specificity, were utilized in order to assess the effectiveness of the model that was proposed.

The results of the trained model's predictions are depicted in Table 2 and Figure 4, which are presented below. The model has been applied to around 2100 tweets from both classes for testing purposes. In the vast majority of instances, there are no tweets that contain sarcasm. A random selection was made before the training procedure in order to choose the data that was tested.

Table 2: Confusion Matrix

		Predicted Class	
		Sarcastic	**Non-sarcastic**
Actual Class	Sarcastic	262	73
	Non-sarcastic	88	1687

The confusion matrix illustrates the performance of the proposed sarcasm detection model. It correctly predicted 262 sarcastic instances and 1687 non-sarcastic instances out of the total 2110 test cases. It has predicted 73 false positives (predicting that an instance is sarcastic when it is actually not). It has also predicted 88 false negatives (predicting that an instance is not sarcastic when it actually is).

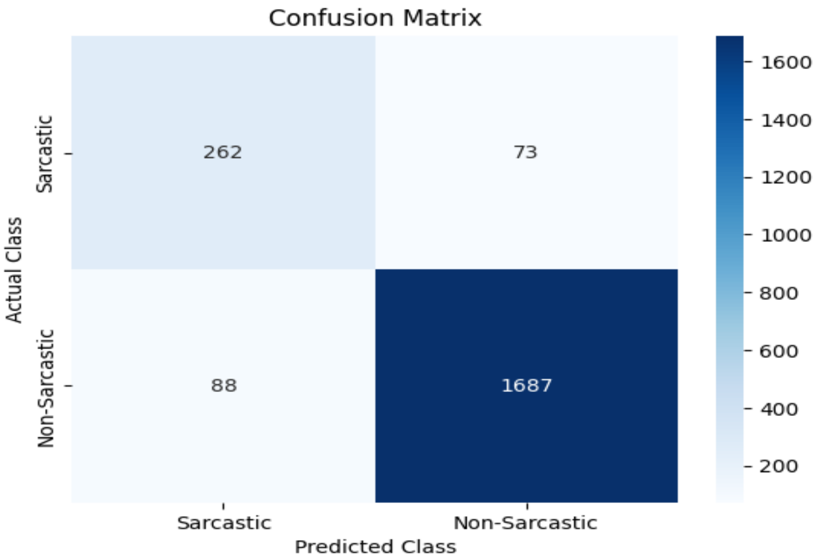

Figure 4: Confusion Matrix

The model is more likely to make false negatives than false positives. Hence, it is implausible that the model will inaccurately predict an instance as sarcastic when it is not. There are more chances to miss a sarcastic instance instead of making wrong predictions. As the sarcasm does not only depend on the text, it also depends on the context, tone of the voice (e.g., all caps for shouting, or using sarcasm-indicative emojis), etc. While in the preprocessing phase, all these features have been removed for better processing of the data and to train the model more effectively. To evaluate the performance of the trained model, this research study has employed several performance measurement metrics. These metrics are based on the data from the confusion matrix. These performance measures help in understanding the model more comprehensively. The performance measures metrics that this research study has employed are precision, recall, F1-score, specificity, and accuracy. The following table (see Table 3) represents the values of these performance measures.

Table 3: Performance Measures

Performance measures				
Precision	**Recall**	**F1-score**	**Specificity**	**Accuracy**
0.7826	0.7486	0.7652	0.9582	0.9246

Precision: This metric expresses how effectively the model identifies true positive instances among all cases predicted as positive. It calculates the ration of the correctly predicted positive cases to the total number of the predicted positives cases. It presents a measure of the positive predication accuracy. In this study, the model achieved a precision of 78.26%. The precisionvalue is computed according to the following formula:

$$Precision = \frac{True\ Positive}{(True\ Positives + False\ Positives)}$$

Recall (Sensitivity): Recall, also known as sensitivity or the true positive rate, measures the model's ability to correctly identify all actual positive instances in a dataset. When the cost of false negatives is high, and capturing the maximum positive cases is important. It is also known as the actual positive rate. In the context of this study, the recall value obtained was 74.86%, which indicates that the model correctly recognized nearly three-quarters of all sarcastic tweets present in the test data. The recall value is computed according to the following formula::

$$Recall = \frac{True\ Positives}{(True\ Positives + False\ Negatives)}$$

F1-score: It provides a balanced evaluation of a model's precision and recall by computing their harmonic mean. This performance measure is important when the distribution of the classes is uneven as in the scenario of this research study. In this study, the developed model achieved an F1-score of 76.52%. The F1-score is determined using the following equation:

$$F1 - score\ = \frac{2 * (Precision + Recall)}{(Precision + Recall)}$$

Specificity: While measuring the performance of a classification model, it is also important to measure the ability of the model to identify the negative instances correctly. Specificity is also known as true negative rate. This performance measure metric calculates the actual negatives that have been correctly predicted by the model. The model has achieved a specificity of 95.82%. The formula to calculate the specificity of the model is as follows:

$$F1 - score\ = \frac{True\ Negatives}{(True\ Negatives + False\ Positives)}$$

Accuracy: This performance measure calculates the ratio of the correctly predicted cases to the total number of test cases. It presents an overall measure of how the model is performing. The model has achieved an overall accuracy of 92.46%. The formula to calculate the accuracy of the model is as follows:

$$Accuracy = \frac{(True\ Positives + True\ Negatives)}{(True\ Positives + True\ Negatives + False\ Positives + False\ Negatives)}$$

The sarcasm detection model developed in this research study has been evaluated using multiple performance evaluation metrics as described above. These metrics help to understand the model performance, such as accuracy. Furthermore, these metrics also aid in understanding other aspects, such as minimizing the false positives or capturing all positive instances, etc.

Table 4 provides a comparison of the various performance measures with existing research. Accuracy is the percentage of all correct predictions. Precision is the percentage of positive predictions that are actually correct. Recall is the percentage of all positive examples that are correctly identified. F1-score is a measure of the overall performance of a model, taking into account both precision and recall. Specificity is the percentage of all negative examples that are correctly identified. Our proposed model outperforms the others in terms of all the performance metrics (where the data is available). The proposed model shows an accuracy of 92.46% and a specificity of95.82%. (Farha & Magdy, 2020; Al-Ghadhban et al., 2017) have provided precision, recall, and F1-score, which are lower than those of the proposed model. Muaad et al. (2022) provide accuracy only, which is 88% in comparison to the accuracy of the proposed model. (Elgabry et al., 2021; Husain & Uzuner, 2021; Israeli et al., 2021) provide accuracy, precision, recall, and F1-score. All these scores are lower than the score of the proposed model.

Table 4: Performance measure comparison with existing research

Model	Accuracy	Precision	Recall	F1-score	Specificity
Proposed Model	0.9246 or 92.46%	0.7826 or 78.26%	0.7486 or 74.86%	0.7652 or 76.52%	0.9582 or 95.82%
Muaad et al, 2022	88%	--	--	--	--
Elgabry et al, 2021	0.7533	0.6858	0.6700	0.5189	--
Husain, & Uzuner, 2021	0.6630	0.6136	0.6318	0.6210	--
Israeli et al, 2021	0.767	0.706	0.702	0.704	--
Farha & Magdy, 2020	--	0.62	0.38	0.46	--
Al-Ghadhban et al, 2017	--	65.9%	71%	67.6%	--

Overall, the proposed model shows the most promising Arabic sarcasm detection results. The proposed model has the highest accuracy and have minimum variation in its performance, while the other models have more variation in their performance. This suggests that the proposed model is more generalizable to different types of Arabic text. The proposed model also has the highest F1-score. This suggests that it is better balanced in terms of precision and recall, which is important for many real-world applications. The specificity of the proposed model is also very high, suggesting that it is unlikely to make false positive predictions, while the specificity is unavailable in the case of the other models.

CONCLUSION

This research study provides a comprehensive overview of the challenges in Arabic sarcasm detection. Key challenges include the lack of benchmarked datasets, class imbalance, and limited NLP resources for Arabic. While sarcastic content is frequently posted online, the scarcity of language resources complicates the development of effective Arabic NLP systems. The absence of large-scale labeled datasets remains a significant obstacle for progress in this area.

To address these challenges, this study presents a model for detecting sarcasm in Arabic text, detailing its architecture, implementation, and evaluation, along with a comparison to existing research. The system is composed of three main modules: preprocessing, training, and testing. The proposed AraBERT transformer-based model was trained and tested on a dataset of approximately 10,500 human-annotated Arabic tweets, with techniques such as class weighting and increased batch sizes applied to manage class imbalance.

The model achieved promising results, with an accuracy of 92.46%, specificity of 95.82%, precision of 78.26%, recall of 74.86%, and F1-score of 76.52%, outperforming existing state-of-the-art approaches. The confusion matrix confirms its ability to predict both sarcastic and non-sarcastic instances correctly.

A limitation of this study is the lack of reported confidence intervals from multiple training runs, which will be addressed in future work to provide a more robust statistical evaluation. Overall, this research highlights the importance of large-scale, labeled datasets for Arabic sarcasm detection and provides a foundation for further studies into Arabic digital communication as both a linguistic and cultural practice.

Funding

This research is not funded.

Author Contributions

This article was sole-authored.

Competing Interests

The author declares no competing interests.

REFERENCES

Alammary, A. S. (2022). BERT models for Arabic text classification: A systematic review. Applied Sciences, *12*(11), 5720. https://doi.org/10.3390/app12115720

Al-Ghadhban, D., Alnkhilan, E., Tatwany, L., & Alrazgan, M. (2017, May 8–10). Arabic sarcasm detection in Twitter. In Proceedings of the 2017 International Conference on Engineering & MIS (ICEMIS) (pp. 1–7). IEEE. https://doi.org/10.1109/ICEMIS.2017.8272990

Alhazmi, S., Khan, S., & Syed, M. H. (2023). Learning-related sentiment detection, classification, and application for a quality education using artificial intelligence techniques. Intelligent Automation & Soft Computing, 36(3), 3487–3499. https://doi.org/10.32604/iasc.2023.036297

Almaaytah, S. A., & Alzobidy, S. A. (2023). Challenges in rendering Arabic text to English using machine translation: A systematic literature review. IEEE Access, 11, 94 772–94 779. https://doi.org/10.1109/ACCESS.2023.3309642

Alqahtani, A., Alhenaki, L., & Alsheddi, A. (2023). Text-based sarcasm detection on social networks: A systematic review. International Journal of Advanced Computer Science and Applications (IJACSA), 14(3). http://dx.doi.org/10.14569/IJACSA.2023.0140336

Antoun, W., Baly, F., & Hajj, H. (2020). AraBERT: Transformer-based model for Arabic language understanding. arXiv preprint arXiv:2003.00104. https://arxiv.org/abs/2003.00104

Bamman, D., Eisenstein, J., & Schnoebelen, T. (2014). Gender identity and lexical variation in social media. Journal of Sociolinguistics, 18(2), 135-160. https://doi.org/10.1111/josl.12080

Bouamor, H., Habash, N., Salameh, M., Zaghouani, W., Rambow, O., Abdulrahim, D., Obeid, O., Khalifa, S., Eryani, F., Erdmann, A., & Oflazer, K. (2018). The MADAR Arabic Dialect Corpus and Lexicon. Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018), 3387-3396. European Language Resources Association. https://aclanthology.org/L18-1535/

Clarke-Dawson, C. (2025). Decoding Digital Culture with Science Fiction: Hyper-Modernism, Hyperreality, and Posthumanism. Journal of Cultural Analysis and Social Change, 10(1). https://doi.org/10.20897/jcasc/16626

Davidov, D., Tsur, O., & Rappoport, A. (2010). Semi-supervised recognition of sarcasm in Twitter and Amazon. In M. Lapata & A. Sarkar (Eds.), Proceedings of the Fourteenth Conference on Computational Natural Language Learning (pp. 107–116). Association for Computational Linguistics. https://aclanthology.org/W10-2914/

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805. https://arxiv.org/abs/1810.04805

Elgabry, H., Attia, S., Abdel-Rahman, A., Abdel-Ate, A., & Girgis, S. (2021). A contextual word embedding for Arabic sarcasm detection with random forests. In N. Habash, H. Bouamor, H. Hajj, W. Magdy, W. Zaghouani, F. Bougares, N. Tomeh, I. Abu Farha, & S. Touileb (Eds.), Proceedings of the Sixth Arabic Natural Language Processing Workshop (pp. 340–344). Association for Computational Linguistics. https://aclanthology.org/2021.wanlp-1.43/

Farha, I. A., & Magdy, W. (2020, May). From Arabic sentiment analysis to sarcasm detecion: The ArSarcasm dataset. In Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection (pp. 32–39). Association for Computational Linguistics. https://aclanthology.org/2020.osact-1.5/

Goswami, B., Bhavsar, N., Alzobidy, S. A., Lavanya, B., Udhayakumar, R., & Rajapandian, K. (2025). Sentiment analysis using natural language processing. In Natural Language Processing for Software Engineering (pp. 283-294). John Wiley & Sons, Inc.

Habash, N. (2010). Introduction to Arabic natural language processing. Morgan & Claypool Publishers.

Hutto, C. J., & Gilbert, E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. Proceedings of the International AAAI Conference on Web and Social Media, 8(1), 216–225. https://doi.org/10.1609/icwsm.v8i1.14550

Israely, R., Itali, R., Khutoretsky, Y., Stern, R., & Schler, J. (2021). Sarcasm and sentiment detection in Arabic tweets using BERT-based models and data augmentation. In Proceedings of the Sixth Arabic Natural Language Processing Workshop (pp. 396–402). https://aclanthology.org/2021.wanlp-1.48

Joshi, A., Bhattacharyya, P., & Carman, M. J. (2017). Automatic sarcasm detection: A survey. ACM Computing Surveys, 50(5), Article 73. https://doi.org/10.1145/3124420

Liu, B. (2012). Sentiment analysis and opinion mining. Morgan & Claypool Publishers.

Loshchilov, I., & Hutter, F. (2017). Decoupled weight decay regularization. arXiv preprint. https://arxiv.org/abs/1711.05101

Muaad, A. Y., Jayappa Davanagere, H., Benifa, J. V., Alabrah, A., Naji Saif, M. A., Pushpa, D., Al-antari, M. A., & Alfakih, T. M. (2022). Artificial intelligence-based approach for misogyny and sarcasm detection from Arabic texts. Computational Intelligence and Neuroscience, 2022, Article 7937667. https://doi.org/10.1155/2022/7937667

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends® in Information Retrieval, *2*(1–2), 1–135. https://doi.org/10.1561/1500000011

Rahma, A., Azab, S. S., & Mohammed, A. (2023). A comprehensive review on Arabic sarcasm detection: Approaches, challenges and future trends. IEEE Access, 11, 93720–93740.

Safaya, A., Abdullatif, M., & Yuret, D. (2020). KUISAIL at SemEval-2020 Task 12: BERT-CNN for offensive speech identification in social media. In Proceedings of the Fourteenth Workshop on Semantic Evaluation (pp. 2054–2059). https://doi.org/10.18653/v1/2020.semeval-1.271

Shapiro, A. N. (2024). Decoding digital culture with science fiction: Hyper-modernism, hyperreality, and posthumanism. Transcript Verlag.

Talafha, B., Za'Ter, M. E., Suleiman, S., Al-Ayyoub, M., & Al-Kabi, M. N. (2021, August). Sarcasm detection and quantification in Arabic tweets [Preprint]. arXiv. https://doi.org/10.48550/arXiv.2108.01425

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention Is All You Need. Advances in Neural Information Processing Systems, 30, 5998–6008. https://arxiv.org/abs/1706.03762

