Journal of Computer Science

Research Paper

AraSenti-MARBERT-DynGCN: An Advanced Framework for Sarcasm Detection in Arabic Text

Bassma M. Mousa[1], Mohammed H. Haggag[2] and Mervat Gheith[1]

[1]*Faculty of Postgraduate Statistical Research, Cairo University, Cairo, Egypt*
[2]*Faculty of Computer Science, Helwan University, Cairo, Egypt*

Article history
Received: 1 December 2024
Revised: 8 February 2025
Accepted: 22 March 2025

*Corresponding Author:
Bassma M. Mousa Faculty of Postgraduate Statistical Research, Cairo University, Cairo, Egypt
Email: b.m.m.othman@gmail.com

Abstract: Sarcasm detection in Arabic texts is challenging because of the complexity in Arabic morphology, the multitudes of dialects used by native speakers, and its strong reliance on context. The lack of standardized orthography in most written forms of Arabic, combined with the frequent use of colloquialisms and metaphorical expressions, makes the task all the more difficult when trying to detect sarcasm. In this paper, we present a hybrid approach for Arabic sarcasm detection by integrating sentiment analysis and contextual embeddings with Dynamic Graph Convolutional Networks (DynGCNs). Our model uses pre-trained language model MARBERT for building rich contextualized word embeddings, AraSenti for accurate sentiment polarity classification, and adopts DynGCNs for capturing syntactic dependencies in the text and updating them dynamically. We also applied state-of-the-art preprocessing to handle informal characteristics such as emoticons and punctuation, which are very necessary in recognizing sarcasm in informal Arabisc speech. The proposed approach was empirically evaluated on two well-known Arabic Sarcasm detection repositories: iSarcasmEval and ArSarcasm-v2 through extensive experiments. As a result, on the iSarcasmEval data split it reached an accuracy of 92.8% and F1-score reaching up to 78.5%, significantly outperforming its counterparts such as AraBERT, QARiB, and MARBERT based models. Its performance on the ArSarcasm-v2 dataset gave an accuracy of 86.5% while the F1-score stood at 71.7%, hence making it a robust method across different datasets.

Keywords: Arabic Sarcasm Detection, MARBERT, Dynamic Graph Convolutional Networks (DynGCNs), AraSenti

Introduction

Sarcasm is a sophisticated linguistic mechanism whose usual meaning is to convey meaning contrary to the literally expressed one, thus it presents a great challenge before both human understanding and computational models. The detection of sarcasm is rather challenging in NLP, especially on informal online platforms such as social media (Mohamed et al. 2024). The Arabic language opens up more challenges because of its rich morphology, diversified dialects, and complicated syntactic structures. This is difficult, as most of the traditional models for sarcasm detection can hardly catch the fine nuances of Arabic sarcasm, considering the widespread use of emoticons, non-standard spellings, and various punctuation patterns that alter the sentiment and meaning of a statement (Rahma et al., 2023). The present study proposes a hybrid approach to such detection by combining sentiment analysis with contextual embeddings and graph networks to overcome those challenges. Our approach employs MARBERT, an Arabic-specific pre-trained model, to extract deep contextual embeddings in textual input. We also introduce sentiment analysis through the AraSenti knowledge base, which measures the sentiment score of texts. DynGCN are applied to identify and represent the changing structural relationships, hence helping the model fit the on-going associations' changes between words and phrases in real time. This dynamic model enables a better grasp of syntax and semantics. It can considerably increase the model capacity to handle different contexts, which becomes pretty handy for this sarcasm detection task. More concentration is paid to the use of emoticons and punctuation, as both are fundamental in sarcasm detection, especially in informal Arabic writing. The next sections describe the three components of the proposed framework in detail.

AraSenti

AraSenti is a sentiment analysis tool developed specifically for Arabic language processing, which utilizes a rich sentiment lexicon combined with state-of-the-art natural language processing techniques that can accurately capture the sentiment conveyed in Arabic texts, both formal and informal, from various dialects. Accounting for the rich morphology and unique linguistic features of Arabic, AraSenti has been found particularly helpful in capturing shifts in sentiment, which very often are crucial to the detection of sarcasm. It provides a very important contribution to enhancing the performance of the model in sarcasm detection, adding more depth to the sentiment understanding that combines with other contextual and syntactic features (Nora et al., 2017).

MARBERT

MARBERT is a pre-trained Language Model (LM) for the Arabic text processing task, catering to the complexities brought in by its rich morphology, complex syntax, and multi-dialects. Built on the BERT architecture, MARBERT has been fine-tuned on a very large corpus of Modern Standard Arabic and dialectal Arabic from social media platforms. It has, therefore, been empowered to capture deep contextual representations of Arabic text that make it very effective in tasks related to sentiment analysis, sarcasm detection, and text classification. By leveraging its capability for handling various forms of the Arabic language, MARBERT yields significant improvements in performance compared to traditional models, especially on tasks with informal or dialectal text (Abdul-Mageed et al., 2020).

DynGCN

DynGCNs extend the traditional graph convolutional networks (GCNs) with dynamic changes in graph structure. In contrast to the static GCNs, which admit fixed graph topologies, DynGCNs model graphs where relationships and interactions are evolving. It is this dynamic nature that allows them to capture the temporal dependencies and spatial interactions that change over time, making them really effective in real-world applications involving time-series data or networks evolving continuously, as in the case of social media or textual data. In the task of Arabic sarcasm detection, DynGCNs prove especially helpful since they are capable of adapting to a fluid or context-dependent nature of sarcasm. Sarcasm in Arabic is greatly context- bound, especially in informal texts, where changes in context, emotional tone, and dynamic interactions between words make dynamic graph structures the right tool to handle such complexities. The model can embrace the dynamic graph representation of words and their syntactic dependencies, adaptation, and learning from the evolving linguistic cues and relations. In this way, it improves the model capability to catch sarcasm in various dialects and informal expressions. This dynamic approach captures in a more fitting manner the shifting contexts and subtle cues defining sarcastic remarks, after all a huge improvement over traditional model. The models with dynamic graph representations, therefore, can accommodate and capture these nuances in the interaction of the Arabic text, which bears metaphors, cultural references, and informal expressions important in sarcasm detection. The advanced capacity, with which the DynGCNs are equipped, increases the prospect of detecting sarcasm in a wide variety of Arabic dialects and also in the informal contexts prevalent in the datasets iSarcasmEval and ArSarcasm- v2 datasets (Li et al., 2020). The key difference between DynGCN and GCN as show in table 1.

The rest of the paper is structured as follows: We conduct a comprehensive related work review in Sect. 2. In Sect. 3, we elaborate on the proposed framework. The detail of configuration and parameter setting is present in Section 4 of this paper. The performance evaluation is given in Sect. 5 and, conclusions and future work are presented in Sect. 6.

Literature Review

In this section, we review related work in sarcasm detection, dividing it into two subsections, one covering Arabic sarcasm detection and the other focusing on English and multilingual Sarcasm Detection sarcasm detection. By exploring both, we try to provide a comprehensive overview of the current state of sarcasm detection across different linguistic contexts.

Arabic Sarcasm Detection

The complexity and variability of dialects place

Arabic at an infant stage in sarcasm detection compared to other languages. Due to the challenges presented, several approaches have emerged concerning Arabic sarcasm detection. Paper (Alakrot et al., 2024) proposes the ArSa-Tweet model, using deep learning for the detection of sarcasm in Arabic tweets. The advanced preprocessing steps have been incorporated into the model to improve the classification performance. They adapted several deep learning models such as LSTM, Multi-headed CNN-LSTM-GRU, BERT, and AraBERT V01/V02. The paper also introduces ArSa- data, an Arabic tweet dataset collected from curated sources. Comparative evaluation reveals that the ArSa-Tweet model, along with the AraBERT-V02 model, shows the best performance with top accuracy on different metrics compared to other approaches. In paper (A.Alakrot et al., 2024) explains how sarcasm detection is important for advancing automated sentiment analysis, an application that has far- reaching use in customer service and product development. Nevertheless, research on sarcasm detection is not much of work done for Arabic, which includes Libyan Arabic dialects. Here a study is working on 5,082 Facebook posts that are labeled as sarcastic or not by experienced Libyan dialect annotators. Then extract syntactic and lexical features were then employed to trigger an SVM model, with which they achieved impressive performance accuracy rates of 79.15 and 79.3% for precision, 79.7% for accuracy, and an F1 score of 79.5%. In paper (Ibrahim et al., 2020) proposes ArSarcasm, an Arabic sarcasm detection dataset, through the reannotation of some existing Arabic sentiment analysis (SA) datasets. It consists of 10,547 tweets, out of which 16% are labeled as sarcastic. Apart from sarcasm, this dataset covers sentiment and dialects. The analysis reveals the subjective nature of these tasks, as annotator biases led to shifts in sentiment labels. Experiments highlight how state-of-the-art sentiment analysis models degrade when handling sarcastic content. A deep learning model based on BiLSTM, trained on this dataset, achieves an F1 score of 46%, underscoring the task difficulty and providing a baseline for future research. In paper (Abdul-Mageed et al., 2021), they have presented their work for developing two powerful Arabic language models inspired by Transformers. Their models are pre-trained on large-to-huge datasets that span different domains and textual genres, including social media. By pre-training ARBERT and MARBERT-v2 on dialectal Arabic, they aim at enabling downstream NLP technologies serving wider and more diversified communities. Top best models outperform (or match) XLMRLarge (~ 3.4× bigger than their models), and hence are more energy-efficient at inference time. Their models considerably outperform AraBERT, currently the best-performing Arabic pre-trained LM. They also presented AraLU, a large and diverse benchmark for Arabic NLU comprising 42 datasets thematically organized into 6 main tasks. clusters. AraLU fills a serious lacuna in Arabic and multilingual NLP and is bound to help drive innovation and enable informative comparisons in the area.

English and Multilingual Sarcasm Detection

Multilingual sarcasm detection has become an immensely complicated, though growing field in NLP, which tries to find out sarcasm across languages, dialects, and code-mixed or code-switched texts. This domain is considered more problematic because the syntactic, semantic, and cultural expression varies from one language to another. Sarcasm usually depends on subtle contextual and cultural cues, which are quite different between languages, making the task of its detection much more laborious than in a multilingual environment. In paper (Pandey et al., 2023) introduced an ensemble approach for sarcasm detection on Reddit and Twitter, presented at the Second Workshop on Figurative Language Processing (ACL 2020). Its ensemble model combined the predictions of the four component models—sarcasm probability—with features such as comment sentiment, length, and platform source. Its components include an LSTM with hashtag and emoji representations; a CNN-LSTM with casing, stop words, punctuation, and sentiment features; an MLP based on infersent embeddings; and an SVM trained on stylometric and emotion-based features. Most of the models take as input the two turns of conversation prior to the response for contextual information, though the SVM does not use any contextual features and only uses features from the response itself. The ensemble created by applying an Adaboost classifier with decision trees as base estimators was also very effective, with F1- scores of 67% on Reddit and 74% on Twitter. It has implications for the effectiveness of combining different linguistic and contextual features and incorporating a variety of model architectures to improve sarcasm detection on social media. The paper (Pandey et al., 2023) achieved by feeding the BERT model for generating embeddings of Sarcasm is a sophisticated linguistic mechanism whose usual meaning is to convey meaning the code-mixed text, which are further passed through an LSTM layer to classify the sentences into sarcastic and non-sarcastic. The experimental results show that the BERT-LSTM model yields high accuracy in capturing

DOI: 10.3844/jcssp.2025.2709.2717

subtle sarcasm in code-mixed text and improvement up to 6% in F1- score over the baseline models. This study (Peng et al., 2024)) focuses on Multi-modal sarcasm detection refers to identifying sarcastic intent in multi-modal inputs by analyzing the associated sentiment. Recent work on vision large language models has brought substantial improvements over a range of multi-modal applications. Inspired by these, they perform a systematic study on the contribution of vision large language models to the zero-shot multi-modal sarcasm detection task. To more comprehensively understand the subtleties of sarcastic utterances, they propose the S3 Agent, a multi-view agent framework for zero-shot multi-modal sarcasm detection with three key considerations: superficial expression, semantic context, and sentiment analysis. Their experiments using the MMSD2.0 dataset, four prompting strategies, and six models achieve state-of-the-art results, with an average improvement over baselines by 13.2%. In paper (Jiaqi et al., 2024) Moreover, they evaluate their method in a text-only sarcasm detection setting, where it also performs better than current baseline methods. Most of the current graph-based methods rely on static networks to model such features, which are not adaptive to different text-image pairings and may incur the loss of important information because of the fixed structure. In order to deal with these problems, some researchers have begun adopting dynamic networks with GCNs to handle variant contexts dynamically; this imparts further flexibility in learning sarcasm- relevant features from text and images. They contribute to this dynamic field with a multimodal sarcasm detection model that integrates GCNs with a Dynamic Network. The GCN part learns object-level incongruities between text-image pairs, while the Dynamic Network asynchronously adjusts to global- level information across modalities. Empirical studies show their method to be superior to state-of-the-art models for sarcasm detection over multiple datasets.

Proposed Framework Architecture

The proposed framework, as shown in the Figure 1 architectural and Figure 2 block diagrams, is a novel deep learning framework for sarcasm detection in Arabic texts. This framework is an integration of sentiment analysis, contextual embeddings, and DynGCNs. Combining AraSenti, an Arabic sentiment analysis tool, with MARBERT, the state-of-the-art transformer-based language model for the Arabic language, is essential to extracting subtle features of sarcasm. DynGCN further upgrades this framework to update graph structures dynamically during training in order to change the token relationship with any changes induced by the syntactic and semantic dependencies. Such flexibility allows the model to better capture the contextual and structural differences involved in sarcasm. This enriches the representation of sarcasm by adding sentiment polarity scores as node features in the graph. DynGCN uses token embeddings from MARBERT and changes their interconnectivity dynamically toward higher accuracy in sarcasm detection.

Data Preprocessing and Cleaning

Data preprocessing is an important phase that has to do with the transformation of raw textual content information into a format apt for the study of machine learning models. The aim of stage preprocessing is to clean and transform the data in such a way that it becomes more plausible and meaningful with regard to NLP tasks. The typical stage preprocessing steps of the proposed framework are highlighted below.

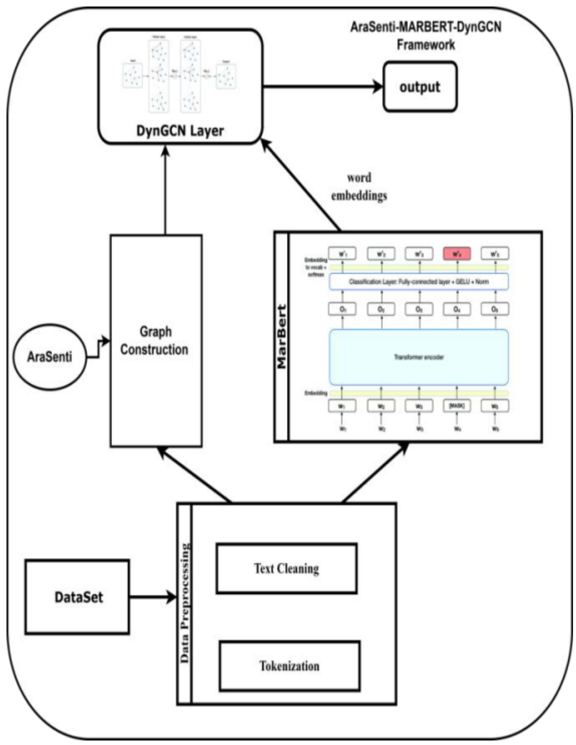

Fig. 1. Proposed Framework Architecture Diagram

Tokenization

Tokenization is one of the important steps in preprocessing before sarcasm detection using MARBERT tokenizer. Tokenization Breaking down the input Arabic text into smaller units, such as words or subwords that the model can process.

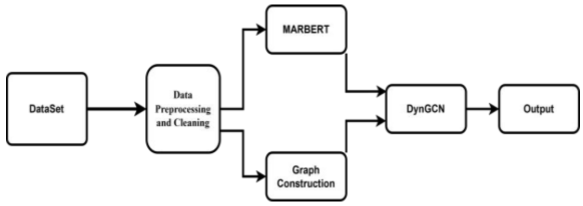

Fig. 2. Proposed Framework Block Diagram

Text Cleaning

Then, we do text cleaning at the preprocessing stage by:

- Remove irrelevant characters and HTML tags, special characters, or other non-text elements.
- Eliminate larger areas or line breaks.
- Remove non-Arabic characters. The text is cleaned of characters that are not Arabic, like numbers, English phrases, and so on.

Embedding Using MARBERT

This phase consists of producing tokens through the MARBERT model that gives contextual embeddings per token. The MARBERT model has been designed primarily for Arabic texts and is very competent at handling the complexities of Arabic morphology and the different dialectal variations. Let:

$T = [t_1, t_2, \ldots, t_n]$ represent the tokenized input text.

Each token t_i is passed through MARBERT to obtain an embedding e_i:

$$e_i = \text{MarBERT}(t_i) \tag{1}$$

The output is a collection of embeddings for all tokens:

$$E = [e_1, e_2, \ldots, e_n] \tag{2}$$

Here, $E \in R^{n \times d}$, where n is the number of tokens and d is the dimensionality of the embeddings produced by MARBERT.

Affective Graph Construction (Adjacency Matrix)

In the proposed framework for Arabic sarcasm detection, constructing the Affective Graph (i.e., the adjacency matrix) plays a crucial role in capturing relationships between tokens based on their sentiment. The graph structure allows the model to represent the semantic and emotional context of the text, is essential for detecting sarcasm.

Sentiment Score Assignment

Given an Arabic token from previous stage for each token, following is Table 2 representation of the sentiment score definition for each token t_i based on the sentiment lookup from the AraSenti lexicon:

Table 2. Sentiment scores are assigned to each token based on the AraSenti lexicon lookup

Token	Sentiment Score
Token has positive sentiment	.01 to 1
Token has negative sentiment	-.01 to -1
Token has neutral or unknown	0

Define Relationship Between Tokens

The adjacency matrix A represents relationships between tokens based on sentiment and position. Let A[i,j] represent the connection between token t_i and token t_j. The weight of the connection can be defined using a sentiment similarity function $f_s(s_i, s_j)$ and a proximity function $f_p(i, j)$.

Sentiment Similarity Function

The Sentiment Similarity Function quantifies the degree of similarity in sentiment between two tokens, t1 and t2. In general, this function is vital to knowing the contextual relationship in tasks like sarcasm detection since the sentiment carried by each token alone may change their interpretation degree inside the text significantly. This similarity function can be mathematically defined and implemented by various techniques. Proposed model apply common approach is to use cosine similarity.

$$F(S_1, S_2) = \frac{S_1 . S_2}{||S_1|| . ||S_2||} \tag{3}$$

Where: $S1 \cdot S2$ is the dot product of the two sentiment vectors. $||S1||$ and $||S2||$ are the magnitudes (Euclidean norms) of the sentiment vectors.

Applying DynGCN

DynGCNs are designed to capture both spatial and temporal dependencies in graph-structured data. If one is going to grasp the meaning of the words due to their relationships in the sentence, this is particularly important to identify the sarcasm well.

Dynamic Graph Representation

The input to the DynGCN is a dynamic adjacency matrix A, which represents the token relationships according to their syntactic dependencies and sentiment scores. This matrix evolves as the context changes.

Feature Matrix Initialization

Each of the token's embeddings from MARBERT make up the first feature matrix $H^{(0)}$. For example, if we have n tokens, we can write this as:

$$H^{(0)} = [E_1, E_2, \ldots, E_n] \quad (4)$$

where E_i is the embedding for token t_i.

Dynamic Graph Update

At each GCN layer, the adjacency matrix A is updated based on the node features $H^{(l)}$ from the previous layer. The updated $A^{(l+1)}$ adjusts edge weights to emphasize tokens more relevant to sarcasm detection dynamically. This is done by computing the similarity score between node embeddings via matrix multiplication, applying non- linearity-ReLU, to filter weak interactions, and softmax normalization to form a probabilistic adjacency matrix whose rows sum to 1. Equation for Dynamic Adjacency Matrix:

$$A^{(l+1)} = \text{softmax}(\text{ReLU}(H^{(l)}W_1 \cdot (H^{(l)} W_2)^T)) \quad (5)$$

Where:

$H^{(l)}$: Node features at layer l.
W_1, W_2: Learnable weight matrices.
ReLU: Activation function ensuring non-linearity.
softmax: Ensures normalized weights for adjacency.

Graph Convolution

Each node aggregates information from its neighbors based on $A^{(l+1)}$, incorporating the dynamic context. Node embeddings are updated at each layer using:

$$H^{(l+1)} = \sigma(A^{(l+1)} \cdot H^{(l)} \cdot W^{(l)}) \quad (6)$$

Where:

σ: Non-linear activation (ReLU).
$W^{(l)}$: Trainable weight matrix.

Final Prediction

$H^{(L)}$ captures sarcasm cues like polarity shifts or contextual mismatch. The pooling layer summarizes token-level sarcasm indicators for final prediction. After L layers, the final node embeddings $H^{(L)}$ are aggregated for classification:

Pooling: Combine node features using mean-pooling:

$$z = \text{Pooling}(H^{(L)}) \quad (7)$$

Output Layer: Predict sarcasm probability:

$$y = \text{softmax}(W_{out} \cdot z + b) \quad (8)$$

Experimental Setup

The proposed system is evaluated using publicly available sarcasm detection datasets, with a primary focus on the ArSarcasm-V2 and iSarcasmEval datasets. The system was developed using Python and important libraries like PyTorch for the deep learning part, Scikit-learn for extra processing and evaluation, and NetworkX in generating and manipulating graphs. It is tested on a machine with an Intel i7 CPU and 16 GB of RAM, and yielded good performance during training and evaluation. The framework itself is optimized using the Adam optimizer; hyperparameters like learning rate and batch size are fine-tuned for stability and efficiency.

Dataset Description

Our proposed framework is based on the iSarcasmEval Arabic tweets dataset. It contains annotations regarding the presence or absence of sarcasm in each tweet. This has been developed for the shared mission of SemEval- 2022 Task 6, which encompassed 3 subtasks in its fold. In the present work, we have utilized the dataset for only one subtask, viz Sub-task A of this shared mission relating to sarcasm detection in Arabic tweets. The iSarcasmEval dataset hosts 4,503 manually labeled tweets, out of which 945 are identified as sarcastic. Its training set consists of 3103 tweets, out of which 745 tweets are sarcastic in nature, amounting to a total of 24%. The test set has 1,400 tweets, out of which 200 are labeled as sarcastic. The table 3 below gives the breakdown of the sarcastic and neutral tweets in the training and testing subsets of the iSarcasmEVAL dataset and ArSarcasm-v2 dataset (Abu et al., 2022). ArSarcasmv2 extends the former ArSarcasm dataset by adding segments of the DAICT corpus and some newly collected tweets. Each tweet was annotated for sarcasm, sentiment, and dialect. The final dataset consists of 15,548 tweets, divided into 12,548 training tweets and 3,000 testing tweets. Results As a part of the shared task on sarcasm detection and sentiment analysis in Arabic, ArSarcasm-v2 was used and released (Abu et al. 2021).

Measures for Evaluation

In the field of machine learning, accuracy and F1 score are two of the most commonly used metrics for evaluating the performance of a model, as explained below.

Accuracy

The proportion of accurately predicted sentences to the total amount of sentences in the system is known as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

where (TP): True positive, (TN): True negative, (FP): False positive, (FN): False negative.

F1 Score

F1 score is calculated by multiplying the accuracy and recall by the total of the precision and recall twice.

$$F1score = \frac{2 * \text{Precision} * \text{Recall}}{Precision + Recall} \tag{10}$$

Where Recall is a measure of how many of the positive cases the classifier correctly predicted, over all the positive cases in the data. Precision measures the exactness of a classifier. A higher precision means less false positives, while a lower precision means more false positives.

Experimental Results

The proposed model performs with a remarkable improvement in accuracy and F1-score for both famous datasets, iSarcasmEval and ArSarcasm-v2, as shown in Tables 3 and 4 and Figures 3 and 4, respectively. These results reveal that leveraging the sentiment features from AraSenti with the dynamic graph convolution abilities of DynGCN provided a clear advantage to our model over all other existing approaches, such as AraBERT, QARiB, and MARBERT. On the other hand, the proposed model achieved an outstanding accuracy of 92.8% and F1- score of 78.5% on the iSarcasmEval dataset compared to the baseline models, showing that it can catch very complex and subtle forms of sarcasm in the Arabic language. On the ArSarcasm-v2 dataset, the model scored an 86.5% accuracy and a 71.7% F1- score, which, although slightly lower than on the iSarcasmEval dataset, still outperforms other models. The consistent performance on different datasets truly shows how robust the proposed model is. The improvements can be attributed to the sentiment scores generated by AraSenti, which add an emotional nuance to the embeddings, and to the dynamic updates of the GCN layers that make the model adaptive to the syntactic and emotional complexities inherent in sarcasm.

Conclusion

In this paper, the proposed framework AraSenti-MarBERT-DynGCN demonstrated significant improvements in Arabic sarcasm detection through the use of effectively exploiting the sentiment analysis obtained from AraSenti, contextual embeddings extracted from MARBERT, and DynGCN dynamically updating the graph structure during training, allowing it to capture evolving relationships between words and emojis. These advantages are further emphasized to show their reflection in the performance evaluation of the

Twitter and TikTok. This would make DynGCN flexible enough by either providing domain knowledge or user-specific contextual graphs. Moreover, the model is optimized for real-time sarcasm detection on a stream of social media content with an efficient trade-off between performance and computational resources.

Acknowledgement

I would like to thank all Editorial Board members who participated in the review and improvement of this work. I would also like to express my gratitude to the publisher for their cooperation in publishing this work.

Author's Contributions

Bassma Mohamed Mousa: Carried out a comprehensive review of the existing literature, contributed significantly to the design of deep learning models, and carried out experiments. She also participated in data collection and preparation.

Mohammed H. Haggag: Offered essential evaluations concerning the methodology employed in the study's design.

Mervat Gheith: Assisted in the interpretation of results and contributed to the editing of the manuscript.

Ethics

This work has not been published before; it is completely original. The corresponding author affirms that there are no ethical problems, and all authors have checked and approved the work.

References

Abdul-Mageed, M., Elmadany, A., & Nagoudi, E. M. B. (2021). ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics* (pp. 7088-7105). Association for Computational Linguistics.

Abu Farha, I., & Magdy, W. (2020). From Arabic sentiment analysis to sarcasm detection: 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). European Language Resource Association.

Abu Farha, I., Oprea, S., Wilson, S., & Magdy, W. (2022). SemEval-2022 Task 6: iSarcasmEval—Intended sarcasm detection in English and Arabic. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)* (pp. 802-814). Association for Computational Linguistics.

Abu Farha, I., Zaghouani, W., & Magdy, W. (2021). Overview of the WANLP 2021 shared task on sarcasm and sentiment detection in Arabic. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop* (pp. 296-305). Association for Computational Linguistics.

Abuein, Q., Al-Khatib, R. M., Jawarneh, A., & Al-Khateeb, M. S. (2024). ArSa-Tweets: A novel Arabic sarcasm detection system based on deep learning model. *Heliyon, 10*, e36892. https://doi.org/10.1016/j.heliyon.2024.e36892

Alakrot, A., Dogman, F., & Ammer, F. (2024). Sarcasm detection in Libyan Arabic dialects using natural language processing techniques. In *Proceedings of the IEEE 4th International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA)* (pp. 761-767). IEEE. https://doi.org/10.1109/MI-STA61267.2024.10599695

Al-Twairesh, N., Al-Khalifa, H., Al-Salman, A., & Al-Ohali, Y. (2017). AraSenti-Tweet: A corpus for Arabic sentiment analysis of Saudi tweets. *Procedia Computer Science, 117*, 63-72. https://doi.org/10.1016/j.procs.2017.10.094

Galal, M. A., Yousef, A. H., Zayed, H. H., & Medhat, W. (2024). Arabic sarcasm detection: An enhanced fine-tuned language model approach. *Ain Shams Engineering Journal, 15*(6), 102736.

Hao, J., Zhao, J., & Wang, Z. (2024). Multi-modal sarcasm detection via graph convolutional network and dynamic network. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24)* (pp. 789-798). Association for Computing Machinery. https://doi.org/10.1145/3627673.3679703

Lemmens, J., Burtenshaw, B., Lotfi, E., Markov, I., & Daelemans, W. (2020). Sarcasm detection using an ensemble approach. In *Proceedings of the Second Workshop on Figurative Language Processing*.

Li, J., Liu, Y., & Zou, L. (2020). DynGCN: A dynamic graph convolutional network based on spatial-temporal modeling. In Z. Huang et al. (Eds.), *Web Information Systems Engineering - WISE 2020* (Lecture Notes in Computer Science, Vol. 12342). Springer. https://doi.org/10.1007/978-3-030-62005-9_7

Pandey, R., & Singh, J. P. (2023). BERT-LSTM model for sarcasm detection in code-mixed social media posts. *Journal of Intelligent Information Systems, 60*, 235-254. https://doi.org/10.1007/s10844-022-00755-z

Rahma, S. S. A., & Mohammed, A. (2023). A comprehensive survey on Arabic sarcasm detection: Approaches, challenges, and future trends. *IEEE Access, 11*, 18261-18280. https://doi.org/10.1109/ACCESS.2023.3247427

Wang, P., Zhang, Y., Fei, H., Chen, Q., Wang, Y., Si, J., Lu, W., Li, M., & Qin, L. (2024). S3 Agent: Unlocking the power of VLLM for zero-shot multimodal sarcasm detection. *ACM Transactions on Multimedia Computing, Communications, and Applications*. https://doi.org/10.1145/3690642

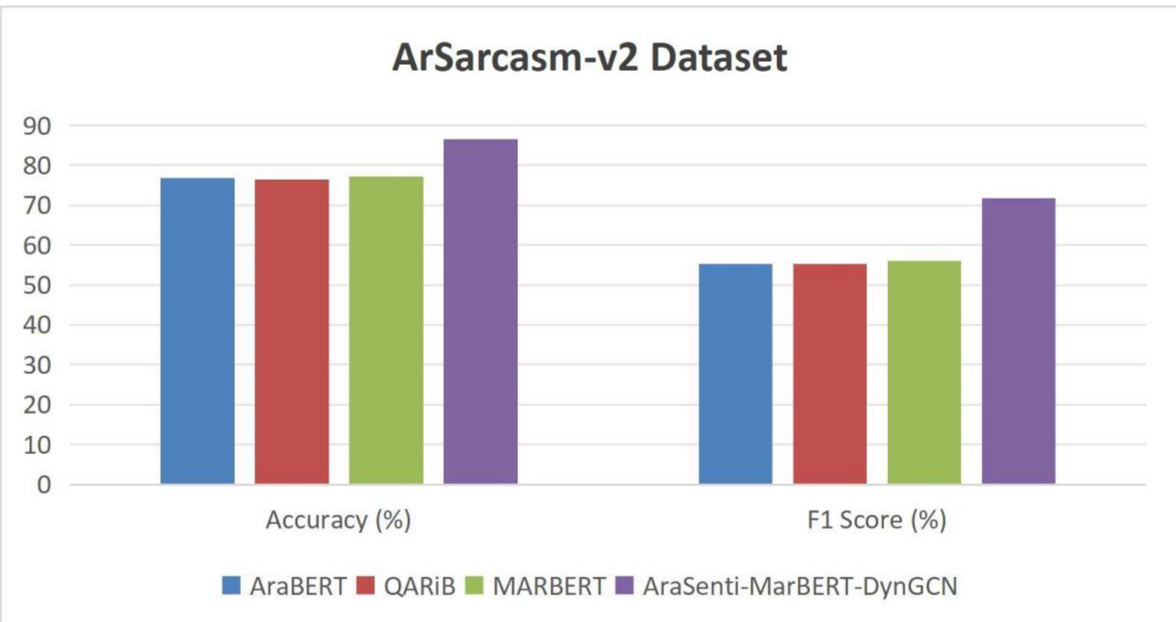

Fig. 3. Performance Metrics for AraSenti-MarBERT-DynGCN Model in iSarcasmEvel Dateset

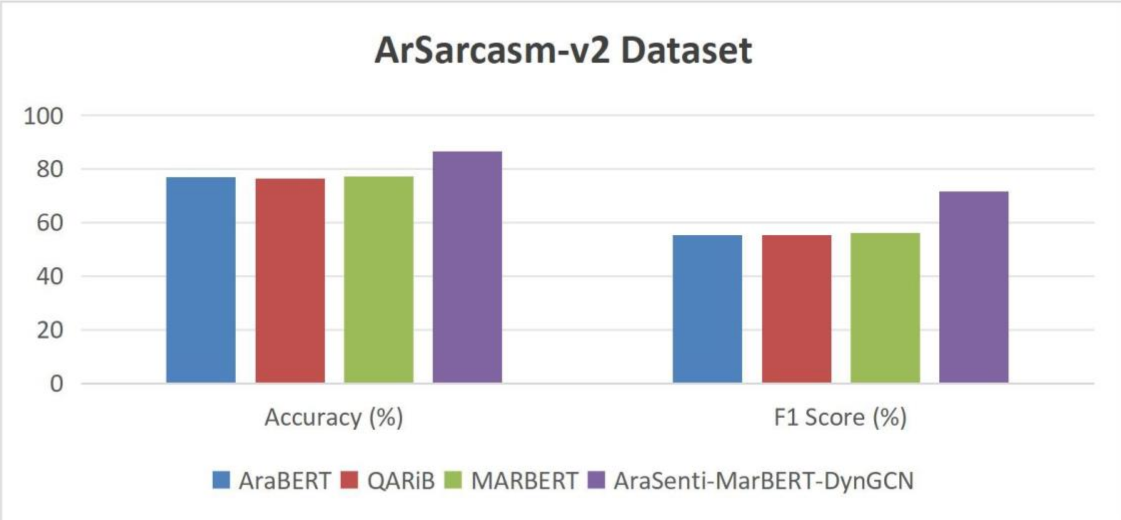

Fig. 4. Performance Metrics for AraSenti-MarBERT-DynGCN Model in ArSarcasm-v2 Dateseset

Table 1. Key Differences Between GCN and DynGCN

Aspect	GCN	DynGCN
Adjacency Matrix	Static	Updated dynamically at each layer based on the token embedding similarity.
Graph Adaptability	it is incapable of adapting to changing contextual dynamics.	Adapts at every layer, enabling dynamic learning of token relationships depending on context.
Complexity	Reduced computational complexity is observed as the adjacency matrix remains constant.	Higher computational cost due to recomputation of the adjacency matrix at each.
Contextual Flexibility	Limited in capturing the evolving relationship in token embeddings across layers.	It effectively models dynamic relationships, so it is well-placed for context-sensitive tasks.

Table 3. Statistics of iSarcasmEVAL and ArSarcasm-v2 datasets

Dataset	iSarcasmEVAL		ArSarcasm-v2	
	Train	**Test**	**Train**	**Test**
Sarcastic	745	200	2,168	821
Neutral	2358	1200	10,380	2179
Total	**3103**	**1400**	**12,548**	**3,000**

Table 4. Performance comparison with baseline

Model	iSarcasmEVAL		ArSarcasm-v2	
	Accuracy (%)	**F1 Score (%)**	**Accuracy (%)**	**F1 Score (%)**
AraBERT (Mohamed et al., 2024)	89.7	63.3	76.9	55.3
QARiB (Mohamed et al., 2024)	88.6	61.9	76.5	55.3
MARBERT (Mohamed et al., 2024)	89.2	63.3	77.1	56.1
AraSenti-MarBERT-DynGCN	**92.8**	**78.5**	**86.5**	**71.7**