ASEAN Journal on Science and Technology for Development

Volume 41
Number 1 *Business Innovation for the Sustainable Development Goals (SDGs) Part 1*

Article 11

2024

Word prediction using Dynamic Skip Connections along with ARABERT and LSTM in Arabic Language

Ahad Almalki
Information Technology Department, Faculty of Computing and Technology, King Abdulaziz University, Jeddah, Saudi Arabia, ayalmalki@stu.kau.edu.sa

Faris Kateb
Information Technology Department, Faculty of Computing and Technology, King Abdulaziz University, Jeddah, Saudi Arabia, fakateb@kau.edu.sa

Rayan Mosli
Information Technology Department, Faculty of Computing and Technology, King Abdulaziz University, Jeddah, Saudi Arabia, rmosli@kau.edu.sa

Recommended Citation
Almalki, Ahad; Kateb, Faris; and Mosli, Rayan (2024) "Word prediction using Dynamic Skip Connections along with ARABERT and LSTM in Arabic Language," *ASEAN Journal on Science and Technology for Development*: Vol. 41: No. 1, Article 11.
DOI: https://doi.org/10.61931/2224-9028.1569
Available at: https://ajstd.ubd.edu.bn/journal/vol41/iss1/11

SPECIAL ISSUE ARTICLE

Word Prediction Using Dynamic Skip Connections Along With ARABERT and LSTM in Arabic Language

Ahad Almalki*, Faris Kateb, Rayan Mosli

Information Technology Department, Faculty of Computing and Technology, King Abdulaziz University, Jeddah, Saudi Arabia

Abstract

Natural Language Generation (NLG) plays a crucial role in modern digital tools, including chatbots, virtual support, content suggestions, and tailored marketing, making bots more responsive and reducing the need for human staff. While there's much research on NLG for languages like English, languages like Arabic, Urdu, and Chinese still face challenges. This study examines Arabic NLG's unique aspects, dialects, and word variations. With around 420 million Arabic speakers globally, it's crucial to advance NLG for this language. We compared three models: Long Short-Term Memory (LSTM), a mix of Bidirectional Encoder Representations from Transformers (BERT) and LSTM, and a version that adds Dynamic Skip Connection (DSC). Our aim is to find the best model for predicting Arabic words with the least mistakes. In our experiment, we found that adding DSC wasn't beneficial. However, combining BERT and LSTM with an attention mechanism reduced loss and favorable perplexity values.

Keywords: Arabic, Natural language generation (NLG), BERT, Long short-term memory (LSTM), Dynamic skip connections (DSC), Deep learning architecture, Transformer models, Arabic language modeling, Attention mechanism, Machine learning, AI

1. Introduction

The field of Natural Language Processing (NLP), a subset of artificial intelligence, has undergone a significant evolution over the past decade, primarily due to advancements in deep learning technologies (Deng & Liu, 2018). These technologies, including transformer models and recurrent neural networks, have considerably enhanced our ability to understand and generate human-like text (Harrag et al., 2021). As a result, we have witnessed substantial progress in numerous applications, from sentiment analysis to automated customer service. Despite these strides, some languages, notably Arabic, still present significant challenges (Ahmed et al., 2022). With its rich linguistic structure and complex syntax, Arabic poses unique hurdles for NLP research. Particularly, Arabic NLG, a discipline within NLP that focuses on producing fluent and contextually appropriate text, proves to be notably challenging (Dukes et al., 2013). The effectiveness of NLG hinges on a comprehensive understanding of the linguistic intricacies and contextual subtleties of the language in question. For Arabic, this task is amplified by factors such as its right-to-left script, numerous dialects, and the

Received 23 October 2023; revised 2 January 2024; accepted 28 February 2024.
Available online 29 May 2024

* Corresponding author.
E-mail addresses: ayalmalki@stu.kau.edu.sa (A. Almalki), fakateb@kau.edu.sa (F. Kateb), rmosli@kau.edu.sa (R. Mosli).

https://doi.org/10.29037/2224-9028.1569

relative scarcity of large, annotated corpora (Masmoudi & Khrouf, 2019). For instance, an Arabic phrase like "حالياً أكتب بحثاً علمياً," which translates to "I am currently writing a scientific research paper," can be written in various dialects or forms and still retain the same meaning. Conventional models may struggle to comprehend these variations, leading to a less accurate understanding and subsequent text generation.

Previous research has sought to navigate these obstacles using various methods. Among them, the deployment of transformer-based models, such as BERT, has shown significant promise (Naous et al., 2021). In the tasks of NLG in languages like English, BERT-based different models have consistently outperformed traditional models (Delobelle et al., 2020). They achieve this by excelling at understanding the contextual meaning of words. However, the effectiveness and adaptability of these models for Arabic NLG, mainly when dealing with less formal or conversational Arabic as found on social media platforms, remain largely unexplored. Addressing these challenges, this study proposes a novel approach: a hybrid model that combines BERT with LSTM networks and DSC. This composite model is expected to overcome the inherent complexities of Arabic NLG by leveraging BERT's understanding of Arabic language, LSTM's capability to capture long-term dependencies, and DSC's ability to skip irrelevant information.

The proposed model is trained and evaluated using the SaudiNewsNet dataset (Al-Hagri [2015], p. 2023 2023). SaudiNewsNet emerges as a paramount dataset for our research due to its richness in content and structure, encompassing over 31,000 Arabic newspaper articles from various Saudi publications. Its comprehensive nature ensures an in-depth exploration of the Arabic language across varied contexts. More importantly, this dataset is not just an accumulation of texts but also integrates essential metadata like source, extraction date, and URL, providing a deeper understanding of the articles' context. This holistic approach of SaudiNewsNet, coupled with the broad spectrum of articles from publications like Al-Riyadh and Al-Jazirah, presents a golden opportunity to rigorously evaluate Arabic NLG models in realistic scenarios. While numerous datasets cater to NLG tasks, the depth and specificity of SaudiNewsNet make it uniquely suited for dissecting the challenges inherent in Arabic NLG. Building on this, the primary objectives of our research pivot around three core areas:

- **LSTM Exploration**: Delving into LSTM's potential in NLG tasks specific to Arabic, discerning its strengths, and recognizing areas demanding refinement.
- **Fusion Approach**: Investigating the combined effects when LSTM and DSC are intertwined with BERT, aiming to unearth model performance and efficiency improvements.
- **Comparative Analysis**: Pitting our novel model against existing benchmarks, critically analyzing its relative advantages, and ensuring it stands up to, if not surpasses, the current gold standards.

Through these objectives, we aspire to sculpt a cutting-edge deep learning model that pushes the boundaries of what is achievable in Arabic NLG. The significance of this research lies in its potential to revolutionize Arabic NLG methodologies. By pioneering a composite model that combines the strengths of BERT, LSTM, and DSC, we aim to provide a solution tailored to Arabic's unique challenges. Arabic, one of the most spoken languages globally, holds immense cultural and practical importance. Yet, the field remains underserved regarding this language's advanced, efficient, and accurate NLG tools. The successful realization of our proposed model could dramatically enhance many applications, streamlining machine translation, powering more context-aware virtual assistants, and fostering richer, more nuanced interactions in AI-driven platforms. Beyond these practical applications, this research is a beacon for future academic pursuits, potentially guiding subsequent investigations and innovations in the broader domain of Arabic NLP.

2. Literature review

The field of NLG has seen a proliferation of methodologies, notably Recurrent Neural Network (RNN) and LSTM. A substantial contribution by (Antoun et al, 2020) involved the application of datasets from Modern Standard Arabic (MSA) and Dialectal Arabic (DA) to adapt the BERT model for the Arabic language. Their experimental results emphasized the efficacy of AraBERT, especially in tasks like sentiment analysis, named entity recognition, and question answering. (Harrag et al., 2021) presented an approach to determine the origin of Arabic sentences, whether human or bot generated. This study demonstrated AraBERT's dominance in accuracy over traditional models like LSTM and GRU. Concurrently, research by (Chouikhi et al., 2021, p. 621) utilized BERT for Arabic sentiment analysis, noting its performance variances across datasets.

Other notable works have explored the diverse applications of Arabic NLP. For instance, (Talafha et al., 2020) documented their work in Arabic dialect identification using labelled tweets. (Elfaik & Nfaoui, 2020) championed the Bidirectional LSTM Network (BiL-STM) for enhanced Arabic NLP. (Haffar et al., 2020) proposed a technique focusing on temporal relationships within Arabic sentences. (Ezen-Can, 2020) work emphasized adapting model selection based on specific tasks and dataset characteristics. In specialized domains of Arabic NLP, (Abboushi & Azzeh, 2023) addressed Arabic poetry generation, introducing AraGPT2. Their work highlighted the potential of this model in generating authentic Arabic poetic verses. (Alyafeai et al., 2023) described "Ashaar 1", a framework geared towards Arabic poetry analysis and generation. On a different note, (Al-Malki & Al-Aama, 2023) steered research towards image captioning specific to Arabic clothing, pioneering a dedicated dataset for this niche. Recognition of Arabic text, both handwritten and printed, has also been a focus area. (Hamida et al., 2023) proposed an innovative approach for recognizing Arabic handwritten text, achieving a near-perfect recognition rate.

Despite a lot of work on the Arabic NLP domain, the language generation literature for Arabic is scarce and needs attention. There are only few works such as Abed & Reiter, 2020; Nagoudi et al., 2023; Shamas et al., 2023), however the results lacks in terms of high false terms generations, low loss and high perplexity.

3. Methodology

This research uses a comprehensive methodology to evaluate the effectiveness of LSTM, BERT, and the combination of BERT, LSTM, and DSC for Arabic NLG. This method starts with carefully preparing the data and analyzing the model's performance.

3.1. Dataset

For this research, we have utilized Saudi Newspapers Corpus (Al-Hagri [2015], p. 2023 2023), known as SaudiNews- -Net, which comprises a collection of 31,030 Arabic newspaper articles complemented by their associated metadata, derived from an assortment of online Saudi newspapers. The data is organized in ZIP files, each named by date, and inside each is a JSON file with an equivalent name. For each article, the JSON structure includes fields such as the source, URL, date of extraction, title, author, and the article content itself. The dataset features articles from a range of Saudi newspapers. For instance, Al-Riyadh newspaper has contributed 4852 articles, followed by Al-Jazirah (3690 articles), Al-Yaum (3065 articles), and so forth. In total, this compilation embodies 8,758,976 words from these articles.

3.2. Model 1- LSTM with self attention

LSTMs are a special kind of RNNs designed to remember long-term dependencies, making them apt for text sequences. They have gating mechanisms (input, forget, and output gates) that regulate the flow of information. Given a sequence $x = (x_1, x_2, \ldots, x_t)$, the LSTM updates its hidden state h_t and memory cell c_t using:

$$f_t = \sigma\left(W_f \,.\, [h_{t-1}, x_t] + b_f\right) \tag{1}$$

$$i_t = \sigma\left(W_f \,.\, [h_{t-1}, x_t] + b_i\right) \tag{2}$$

$$\grave{c_t} = tanh\ tanh\ (W_c \,.\, [h_{t-1}, x_t] + b_c) \tag{3}$$

$$c_t = f_t \times c_{t-1} + i_t \times \grave{c_t} \tag{4}$$

$$o_t = \sigma\left(W_f \,.\, [h_{t-1}, x_t] + b_o\right) \tag{5}$$

$$h_t = o_t \times tanh\ tanh\ c_t \tag{6}$$

While f_t is the forget gate's activation, i_t is the input gate's activation, o_t Is the output gate's activation, $\grave{c_t}$ is the cell's candidate value, and σ represents the sigmoid function. Post the LSTM's sequence outputs, a dropout layer for regularization is used, followed by a dense layer mapping to the desired vocabulary size. This facilitates the prediction of the following possible tokens in the sequence.

3.2.1. Self-attention mechanism

The self-attention mechanism allows the model to consider other words in the input when encoding a word. The weightage assigned to words is calculated using scaled dot-product attention. Given a query Q, a key K, and a value V, the self-attention mechanism is calculated as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{7}$$

Where d_k is the dimension of the key. This ensures that the dot products do not grow too large.

3.2.2. Parameters and settings

- $Learning\ Rate = 0.001$
- $Number\ of\ Epochs = 50$
- $Batch\ Size = 256$

- *Loss* = Cross Entropy Loss
- *Optimizer* = Adam

3.3. Model-2: BERT combined with LSTM

This model leverages the power of a Transformer-based architecture combined with a classification layer. At its core, it uses the same pre-trained Arabic BERT model ('aubmindlab/bert-base-arabertv02'), the foundation for feature extraction. This ensures robust representations of Arabic tokens. These features are then fed into a dense layer with a softmax activation function that helps in the classification task.

3.3.1. Bidirectional Encoder Representations from Transformers

BERT is pre-trained bidirectionally, considering the entire context of a word. During pre-training, BERT uses a masked language model objective. A portion of the input tokens is randomly masked, and the aim is to predict the original vocabulary id of the masked word based only on its context. Given an input sequence x_1, x_2, $_{\dots,}$ x_T, the likelihood is:

$$L = \sum_{t=1}^{T} logP(x_1, \dots, x_{t-1}) \quad (8)$$

The self-attention mechanism weights tokens differently based on the context. Instead of performing a single set of attention operations, transformers use multiple sets, or 'heads'. Given h heads, the multi-head attention is defined as:

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_h)W_O \quad (9)$$

Where: $head_i = Attention(QW_{Qi}, KW_{Ki}, VW_{Vi})$. Here, W_{Qi}, W_{Ki}, and W_{Vi} are parameter matrices and W_O is the output transformation.

3.3.2. Parameters and settings

- *Learning Rate* = 0.001
- *Number of Epochs* = 50
- *Batch Size* = 4
- *Loss* = Cross Entropy Loss
- *Optimizer* = Adam

3.4. Model-3 Arabic language model with DSC

Model 3 is a powerful hybrid that combines recurrent neural networks (RNNs) with the efficacy of pre-trained transformers. It uses the model 'aubmindlab/bert-base-arabertv02' to extract rich representations of Arabic tokens. In order to guarantee that the model can capture long-term relationships in sequences, these are then transferred to an LSTM layer. This combination enables it to operate as a language model by helping to predict following tokens for a given sequence. The transformer uses self-attention mechanisms, just like the earlier models, to record the contextual links among words in the Arabic text.

3.4.1. Dynamic skip connections

The problem of training deeper models is addressed by Dynamic Skip Connections, or DSC, which allow the model to adaptively employ or bypass particular layers during the forward and backward passes. In essence, the DSC lets the network "skip" some levels by allowing it to go around them. When there's a chance of overfitting or when some layers aren't adding much to the learnt features, this can be helpful. Through adaptive learning on when to skip, the network may figure out the best information flow channel.

- *Learning Rate* = 0.001
- *Number of Epochs* = 50
- *Batch Size* = 32
- *Loss* = Cross Entropy Loss
- *Optimizer* = Adam
- *Scheduler* = *torch.optim.lr_scheduler.StepLR(optimizer, step_size* = *1, gamma* = *0.1)*

4. Results/findings

The first model used self-attention processes in conjunction with a conventional LSTM. By incorporating self-attention layers, the model was able to understand contextual dependencies in the dataset. The model proved to be dependable in three separate runs, regardless of batch size fluctuations. Remarkably, the perplexity scores demonstrate the model's ability to produce precise predictions, with validation values of 20.0733 and training values of 20.2483. Additionally, the model's strong training stability is demonstrated by the consistency of training loss over several runs, with training loss at 3.0081 and validation loss at 2.9994 after 50 epochs. The perplexity curves and training and validation losses are depicted in the graph Fig. 1.

The second model combining LSTM and BERT structures demonstrated impressive perplexity scores of 1.23 across three consecutive runs. Despite an increased batch size in the second run, the model maintained a similar perplexity score, with slightly elevated training loss suggesting overfitting. The model's effectiveness in generating coherent Arabic natural language was confirmed by a training loss of

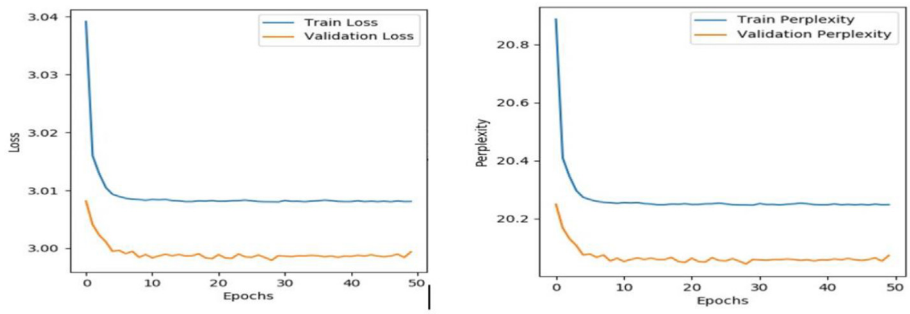

Fig. 1. Results of LSTM model.

6.3929 and a validation loss of 2.1995 after 50 epochs in the below graph, Fig. 2.

The third model integrated Dynamic Skip Connections (DSC) with a transformer architecture to optimize BERT output. The model's perplexity score indicates uncertainty in predictions due to the dynamic nature of skip connections and transformers. The training and validation loss was substantial, with a training loss of 6.7143 and a validation loss of 4.1193 after 50 epochs. The perplexity values for training and validation were 824.12 and 36,113,408.74 respectively, highlighting the unique challenges of this model architecture in the graph Fig. 3.

5. Discussion

To further our understanding of model performance and complexity in the field of natural language processing (NLP), we compare three different models: LSTM, LSTM with AraBERT, and LSTM with AraBERT and DSC. This comparison investigation revealed that the LSTM model, with its recurrent neural network design, performed exceptionally well. This model demonstrated the lowest training and validation losses as well as the best perplexity values, indicating a strong capacity to identify complex patterns in the data. The LSTM model's ability to generalize well to the validation set highlights how well it captures the subtleties that are intrinsic to the language data.

The evaluation of three natural language processing models - LSTM, LSTM with AraBERT, and LSTM with AraBERT and DSC. The LSTM model showed superior performance, with the lowest training and validation losses and the most favorable perplexity values. However, the LSTM with the AraBERT model had higher perplexity values, suggesting

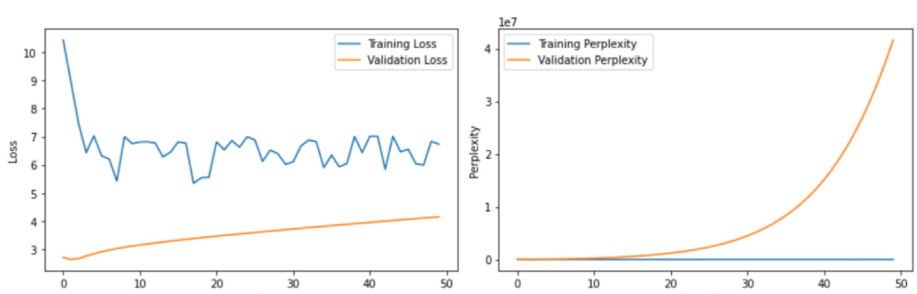

Fig. 2. Results of BERT and LSTM model.

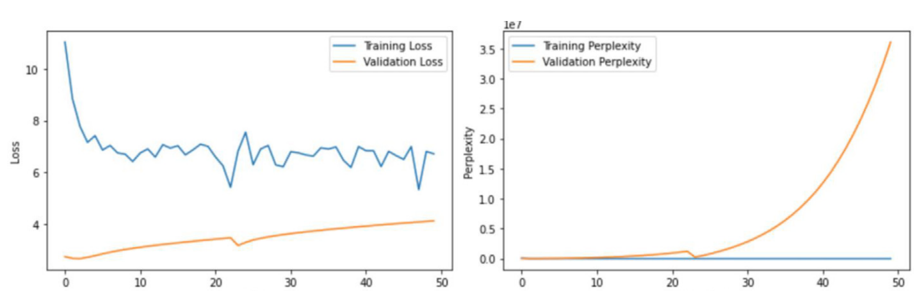

Fig. 3. Results of dynamic skip connection model with LSTM and BERT.

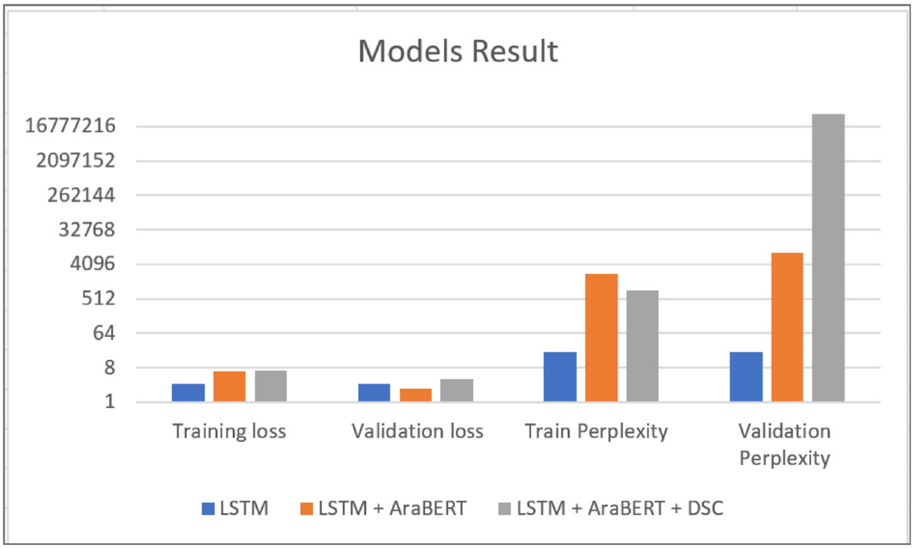

Fig. 4. Comparison of all models and their results.

potential interpretability and model complexity issues. Despite incorporating additional resources, the LSTM with AraBERT and DSC model had the highest validation loss and the highest perplexity, suggesting overfitting or architectural issues. In conclusion, the LSTM model was the most effective and balanced choice explained further in Fig. 4.

6. Conclusion

In conclusion, this research has offered a thorough analysis of three different natural language processing models: the LSTM, LSTM with AraBERT, and LSTM with AraBERT and DSC. With the lowest training and validation losses and the best perplexity values, the LSTM model performed better than the others. Its proficiency in capturing the complex patterns inherent in the language data was demonstrated by its effective generalization to new data. However, the addition of AraBERT and DSC complicated the model in a way that might have improved semantic comprehension but did not enhance performance proportionately, causing overfitting problems in the latter model. The significance of finding a careful balance between model complexity and performance is shown by this study.

Conflict of interest

The authors declare no conflict of interest.

References

Abboushi, O., & Azzeh, M. (2023). Toward fluent Arabic poem generation based on fine-tuning AraGPT2 transformer. *Arabian Journal for Science and Engineering*, 1–13.

Abed, W., & Reiter, E. (2020). Arabic NLG Language functions. In *Proceedings of the 13th international conference on Natural Language Generation* (pp. 7–14).

Ahmed, I. A., Al-Aswadi, F. N., & Noaman, K. M. G. (2022). Arabic knowledge graph construction: A close look in the present and into the future. *Journal of King Saud University-Computer and Information Sciences, 34*(9), 6505–6523.

Al-Hagri. (2015). *Mazen.* (p. 2023). Saudi Newspapers Corpus (SaudiNewsNet).

Al-Malki, R. S., & Al-Aama, A. Y. (2023). Arabic captioning for images of clothing using deep learning. *Sensors, 23*(8), 3783.

Alyafeai, Z., Al-Shaibani, M. S., & Ahmed, M. (2023). *Ashaar: Automatic analysis and generation of Arabic poetry using deep learning approaches*. arXiv Preprint arXiv:2307.06218.

Antoun, W., Baly, F., & Hazem, H. (2020). *Arabert: Transformer-Based model for Arabic Language understanding*. arXiv Preprint arXiv:2003.00104.

Chouikhi, H., Chniter, H., & Jarray, F. (2021). Arabic sentiment analysis using BERT model. In *Advances in computational collective intelligence: 13th international conference, ICCCI 2021, kallithea, rhodes, Greece, september 29–october 1, 2021, proceedings 13*. Springer.

Delobelle, P., Winters, T., & Berendt, B. (2020). *Robbert: A Dutch roberta-based language model*. arXiv Preprint arXiv:2001.06286.

Deng, L., & Liu, Y. (2018). A joint introduction to Natural Language Processing and to deep learning. *Deep Learning in Natural Language Processing*, 1–22.

Dukes, K., Atwell, E., & Habash, N. (2013). Supervised collaboration for syntactic annotation of quranic Arabic. *Language Resources and Evaluation, 47*, 33–62.

Elfaik, H., & Nfaoui, El H. (2020). Deep bidirectional LSTM network learning-based sentiment analysis for Arabic text. *Journal of Intelligent Systems, 30*(1), 395–412.

Ezen-Can, A. (2020). *A comparison of LSTM and BERT for small Corpus*. arXiv Preprint arXiv:2009.05451.

Haffar, N., Hkiri, E., & Zrigui, M. (2020). Using bidirectional LSTM and shortest dependency path for classifying Arabic temporal relations. *Procedia Computer Science, 176*, 370–379.

Hamida, S., Cherradi, B., El Gannour, O., Raihani, A., & Hassan, O. (2023). Cursive Arabic handwritten word recognition system using majority voting and K-nn for feature descriptor selection. *Multimedia Tools and Applications*, 1–25.

Harrag, F., Debbah, M., Darwish, K., & Ahmed, A. (2021). *Bert transformer model for detecting Arabic GPT2 auto-generated tweets*. arXiv Preprint arXiv:2101.09345.

Masmoudi, A., & Mariem Ellouze Khmekhem, Khrouf, M., & Lamia Hadrich Belguith. (2019). Transliteration of arabizi into Arabic script for Tunisian dialect. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP), 19*(2), 1–21.

Nagoudi, El M. B., Ahmed El-Shangiti, Elmadany, A. R., & Abdul-Mageed, M. (2023). *Dolphin: A challenging and diverse benchmark for Arabic NLG*. arXiv Preprint arXiv:2305.14989.

Naous, T., Antoun, W., Mahmoud, R. A., & Hazem, H. (2021). *Empathetic BERT2BERT conversational model: Learning Arabic Language Generation with little data*. arXiv Preprint arXiv: 2103.04353.

Shamas, M., El Hajj, W., Hazem, H., & Shaban, K. (2023). Metadial: A meta-learning approach for Arabic dialogue generation. *ACM Transactions on Asian and Low-Resource Language Information Processing, 22*(6), 1–21.

Talafha, B., Ali, M., Za'ter, M. E., Seelawi, H., Tuffaha, I., Samir, M., Farhan, W., Hussein, T., & Al-Natsheh. (2020). *Multi-dialect Arabic bert for country-level dialect identification*. arXiv Preprint arXiv:2007.05612.