Babylonian Journal of Artificial Intelligence
Vol. 2024, **pp**. 46–53
DOI: https://doi.org/10.58496/BJAI/2024/007 ISSN: 2958-6453
https://mesopotamian.press/journals/index.php/BJAI

Research Article

A Brief Review on Preprocessing Text in Arabic Language Dataset: Techniques and Challenges

Ahmed Adil Nafea[1,*], Muhmmad Shihab Muayad[2], Russel R Majeed [3], Ashour Ali [4], Omar M. Bashaddadh[4], Meaad Ali Khalaf [5], Abu Baker Nahid Sami [6], Amani Steiti[7]

1 Department of Artificial Intelligence, College of Computer Science and IT, University of Anbar, Ramadi, Iraq
2 Department of Computer Networking Systems, College of Computer and Information Technology University of Anbar, Anbar, Iraq
3 College of Education for Pure Sciences, University of Thi-Qar, Thi-Qar, Iraq
4 Center for Artificial Intelligence Technology (CAIT), Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), Bangi, Selangor, Malaysia
5 Department of computer science, AUL University, Beirut, Lebanon
6 Department of Computer Science, University of Anbar Ramadi, Iraq
7 Department of Computer Systems And Networks, Faculty of Information Engineering, University Tishreen, Latakia, Syria.

ARTICLE INFO

Article History
Received 09 Feb 2024
Accepted 27 Apr 2024
Published 18 May 2024

Keywords
Arabic language
Artificial Intelligence
Preprocessing
Natural Language Processing (NLP)
Deep Learning
Machine Learning

ABSTRACT

Text preprocessing plays an important role in natural language processing (NLP) tasks containing text classification, sentiment analysis, and machine translation. The preprocessing of Arabic text still presents unique challenges due to the language's rich morphology, complex grammar, and various character sets. This brief review studied various techniques utilized for preprocessing Arabic text data. This study discusses the challenges specific to Arabic text and current an overview of key preprocessing steps including normalization, tokenization, stemming, stop-word removal, and noise reduction. This survey analyzes preprocessing techniques on NLP tasks and focus on current research trends and future directions in Arabic text preprocessing.

1. INTRODUCTION

Preprocessing text is important step in NLP tasks because it allows for the cleaning and transformation of raw text data into a format that is suitable for analysis in Arabic datasets [1],[2]. The Arabic language is unique a Semitic language, possesses definite characteristics and challenges that need specific preprocessing techniques [3],[4]. These challenges including processing diacritics, spellings, abbreviations, and dialectal variations [5],[6],[7].

The Raw text data usually contains noise, inconsistencies, and variations that can hinder the accuracy and performance of NLP tasks via preprocessing the text, researchers and practitioners can address these issues and create a standardized representation of the data, easing more effective analysis and modelling [8],[9],[10],[11].

There are a lot of issues in text preprocessing of Arabic language characteristics. The language is rich morphology which means the words can submit to significant changes in form based on their grammatical functions and contextual factors [12],[13],[14]. The Arabic language have a complex grammar with syntactic structures. These characteristics needed to utilize focused preprocessing techniques for process the morphological and syntactic complexities of the Arabic language [15], [16].

Corresponding author. Email: ahmed.a.n@uoanbar.edu.iq

In Arabic Language the difference Dialectal shows complicated text preprocessing. Arabic Language dialects are different in pronunciation, vocabulary, grammar, and even word meanings [17],[18],[19]. It is making challenging to process and analyze text across another dialects because these variations within countries and across regions [20],[21]. It is important to get preprocessing techniques to process these dialectal differences to ensure accurate and reliable results in NLP tasks [22].

In this brief review have studied diffrent preprocessing methods for Arabic Language text focusing on the challenges and the solutions proposed in the literature and discuss important preprocessing steps like text normalization, diacritic restoration, spell correction, abbreviation handling, and stemming. This study highlights the effect of preprocessing on Arabic Language NLP tasks like text classification, sentiment analysis, named entity recognition and machine translation. This work focus on address issues related to dialectal variations in Arabic text and highlight the important of building standardized Arabic resources. This review aims to helping researchers in preprocessing Arabic text and improving the accuracy and performance of NLP applications in the Arabic language.

2. Preprocessing Techniques

Preprocessing techniques are important in NLP tasks. The quality of preprocessing have a direct impact on the performance and precision of NLP tasks such as text classification and sentiment analysis. This review studied different preprocessing strategies including normalization, tokenization specifically for Arabic language, stemming and lemmatization, and stop-word removal as shown in Figure 1.

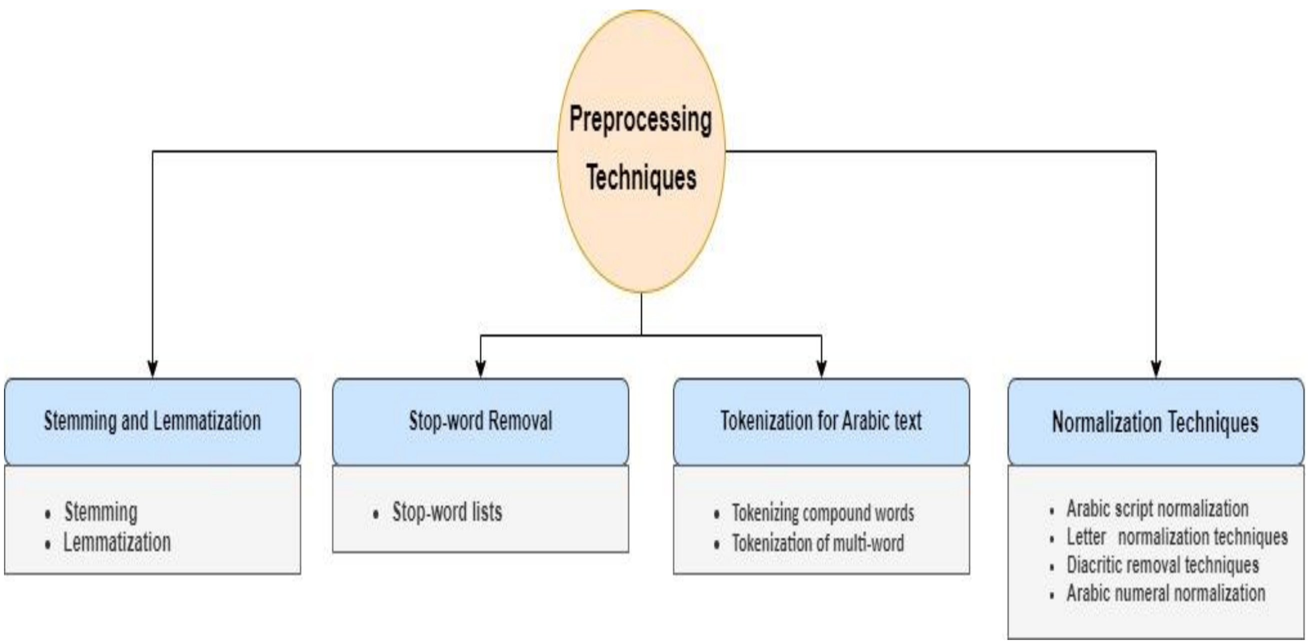

Fig. 1. Preprocessing Techniques in Arabic Texts

2.1 Normalization Techniques

1. **Arabic script normalization:** It applied to changing text to a standardized form by removing ligatures and special characters, and converting non-standard Arabic fonts to standard Unicode representation [23], [24].

2. **Letter normalization techniques:** It works in changing on letter forms and ligatures, ensuring smooth processing of several initial, medial, final, single, and ligature-common letter combinations [25].

3. **Diacritic removal techniques:** are removing vowel sounds and other linguistic features from Arabic text, simplify processing and analysis while protecting accordant information through rule-based approaches or machine learning methods [26].

4. **Arabic numeral normalization** is important for maintaining consistency and compatibility with NLP algorithms operating on Western numeral systems, as it converts Arabic numerals into their equivalent Western counterparts [27].

Table 1 Shows examples of texts before and after applying normalization techniques.

TABLE I. EXAMPLES OF TEXTS AFTER APPLYING NORMALIZATION TECHNIQUES.

Normalization Technique	Original Text	Normalized Text	Notes
Arabic script normalization	من المنصات قبل	من المناسبات قابل	Sometimes the mistake happens in the Arabic script process, Here the text is corrected to the right one.
Letter normalization techniques	آسف بـــــداية	أسف بداية	Changes are made to letter forms and ligatures to ensure smooth processing of various letter combinations. Here, the isolated hamza (آ) is replaced by its standardized form (أ)
Diacritic removal techniques	كَتَبْتُ سَعِيدٌ	كتبت سعيد	The diacritic marks, such as the Fatha ـَ and Damma ـُ, are removed from the text, simplifying it
Arabic numeral normalization	256 31415927مم	256 31415927	The Arabic numerals are converted into their equivalent Western numerals

2.2 Tokenization for Arabic text

Tokenization rules are used to split text into tokens and consider changes from prefixes and suffixes. It's challenging due to morphological complexity and clitics [28],[29].

1. **Tokenizing compound words**: is to identify and treat them as separate tokens utilizing rule-based or statistical methods, leveraging patterns and linguistic resources [30].

2. **Tokenization of multi-word**: utilizing lexicons or specialized resources and statistical models to preserve meaning and avoid incorrect segmentation [30].

2.3 Stemming and Lemmatization

1. **Stemming** is a method of reducing words to their base form. It is an important aspect of Arabic linguistics, aiding in text analysis and retrieval tasks [31],[32],[33].
2. **Lemmatization** is an important process in Arabic, determining the base or dictionary form of words, accounting for morphological changes in lemmas based on root, pattern, and linguistic features [34], [35].

Table 2 illustrates the Stemming and Lemmatization processes with suitable examples of these techniques.

TABLE II. AN EXAMPLE OF STEMMING AND LEMMATIZATION & STOP-WORD REMOVAL PROCESS

Technique	Original Text	Normalized Text	Notes
Stemming	تعلمت	علم	The suffix "ت" is removed, resulting in the stem "علم".
Lemmatization	كتابات	كتاب	The base form "كتاب" is the root form of the word and represents its base dictionary form
Stop-word removal	في هذا اليوم الجميل، ذهبت إلى المكتبة لشراء كتاب جديد عن العلم	هذا اليوم الجميل، ذهبت المكتبة شراء كتاب جديد العلم	Stop-word list here is ["في", "إلى", "ل", "عن"]

2.4 Stop-word Removal

Stop-word lists: They contain common words with little semantic value. It is utilized in text preprocessing to decrease noise and improve the efficiency of downstream NLP tasks [36],[37]. Figure 3 shows an example of Stop-word Removal process.

3. Preprocessing on NLP Tasks

1. **Text classification:** the preprocessing methods such as normalization, tokenization, noise reduction, and handling dialectal differences boost the accuracy of Arabic text categorization. This can improve the extraction of features and the accuracy of the model [38].
2. **Sentiment analysis:** it is important to apply preprocessing normalization techniques, and appropriate strategies for processing negation and Arabic-specific idioms [39].
3. **Named entity recognition:** the preprocessing techniques like tokenization, normalization, and dialectal differences enhance Arabic named entity recognition (NER) tasks, improving precision and recall in identifying and categorizing entities accurately [40].
4. **Machine translation:** Preprocessing in Arabic including normalization, tokenization, stemming, and dialectal variations, enhances translation accuracy by ensuring alignment between source and target text[41].

4. Evaluation Metrics for Preprocessing Techniques

The evaluation of preprocessing techniques for the Arabic Language Datasets is essential as it may impact the model's input data quality and thus influence the quality of the results emerging from the model. Rich morphology and dialectic variations in Arabic pose unique challenges. [36],[42],[43]. The main metrics and techniques to measure the effectiveness of preprocessing in Arabic language data handling are given below:

1- Data Quality Metrics:

The quality of the dataset can be assessed along several dimensions, either before or after preprocessing:

i. ***Coverage:*** refer to the ratio of successfully processed text (e.g., tokenized, normalized) to the total required to be processed. High coverage is the indicator that preprocessing tools consider various aspects of language in an effective way [19].

ii. ***Accuracy:*** Stemming, lemmatization, part of speech, and other preprocessing activities performed should be checked for accuracy against manually annotated referential standards to act as a gold standard[44].

iii. ***Consistency:*** Preprocessing must be consistent for the given text and more so in handling language variants and dialects[45].

2- Downstream Task Performance Metrics:

This can justify the need for preprocessing by its effect on these downstream tasks. Commonly used metrics are [46]:

a. ***F1 Score, Precision, and Recall:*** These measures can help to quantify the balance of finding relevant cases and keeping a low number of false positives for tasks like Named Entity Recognition (NER) and Sentiment Analysis (SA).

b. ***Word Error Rate (WER):*** It is the rate of incorrect predictions of complete words by the hypothesis with respect to the sum of words in the reference.

c. ***BLEU, ROUGE, METEOR** and BERTScore**:*** Metrics to measure the degree of correspondence between the pre-processed output and reference translations or summaries in machine translation and text summarization.

3- Computational Efficiency:

According to Said et al. [47] Efficiency here is key, especially in deployment cases.

i. ***Processing Time:*** Time taken when applying preprocessing steps. Prefer faster processing but with minimal loss in accuracy.

ii. ***Resource usage:*** Monitor CPU and memory usage during preprocessing to ensure how NLP applications can scale up.

4- Specificity to Preprocessing Arabic language:

Here are some particularly relevant for Arabic according to Guellil et al. [48]:

- ***Dialect Handling:*** Evaluate how preprocessing techniques handle the variation of Arabic dialects. The criteria for this can either be the completion of a task or dataset that strictly relies on dialect, or you could have an artificially created mixed dataset.

- ***Script Normalization:*** Arabic script can be written in variant orthographies. This should handle such variants uniformly to be effective in preprocessing.
- ***Clitic handling:*** Arabic has wide usage of clitics. This means the accurate segmentation with reattachment of clitics is to be done without semantic loss of meaning.

5- User-Based Evaluation

In some cases, human judgment through user studies may reveal the practical usability of preprocessing steps, especially for application purposes like translation or content moderation, where user perception is key [49].

6- Error Analysis:

These are the types of error-specific analyses that need to be carried out to know which types of errors are introduced or corrected by the preprocessing. This will throw light on the further refinements to be applied to preprocessing techniques [50].

Each of the approaches to evaluation provides some angle upon the efficiency of preprocessing and taken together, can serve to further refine the technique for better serving Arabic language processing specific needs. They constitute a broad-range assessment that may lead to the improvement of preprocessing methods, which finally results in more robust and accurate NLP applications.

5. Current Research Trends and Future Directions

Deep learning techniques such as neural networks and transformer models, are being applied to Arabic text preprocessing to improve accuracy [51],[52]. Dialectal variations in Arabic pose significant challenges, and ongoing research focuses on developing techniques to handle these variations effectively [53]. This includes building resources for dialectal Arabic, developing dialect-aware preprocessing algorithms, and exploring machine learning approaches that adapt to dialectal differences. The standardized Arabic Language sources like morphological analyzers, lexicons and annotated corpora, are important for efficient text preprocessing [54]. The future research aims to develop comprehensive, high-quality resources covering different dialects, variations, and fields of Arabic for more accurate and consistent preprocessing techniques.

6. Challenges in Arabic Text Preprocessing:

Arabic Language is a tough language having a rich word structure, needing strong algorithms for tokenization, stemming, and lemmatization. There are many problems in text preprocessing depending on dialectal differences in social media and online communication. The Arabic language is a lacks widespread linguistic sources like stop word lists and language models, delaying the improvement of preprocessing techniques. Discretization and vocalization signs are usually lost in informal texts, so it complicates the process. Researchers are always performing modern techniques to develop Arabic Language text preprocessing accuracy.

7. Conclusion

Text preprocessing is important for tasks of NLP Arabic Language via addressing the issues shown with Arabic language. The methods for normalization, tokenization, stemming, stop-word removal, and noise reduction facilitate in cleaning and transforming Arabic Language text into a appropriate format for analysis and modelling. The preprocessing is important to effect on several NLP tasks like text classification, sentiment analysis, named entity recognition, and machine translation. The evaluation metrics and performance comparison benefits in evaluating the efficiency of preprocessing techniques, while current research trends focus on deep learning approaches with handling dialectal differences, and building standardized Arabic resources. By advancing preprocessing techniques, researchers and practitioners can improve the accuracy and performance of Arabic Language NLP applications and enable visible access to Arabic language processing technologies.

Conflicts Of Interest

The authors don’t have any conflict of interest in regards of this research.

Funding

The author's paper explicitly states that the research was self-funded and no support was received from any institution or sponsor

Acknowledgment

The author acknowledges the support and resources provided by the institution in facilitating the execution of this study.

References

[1] M. O. Hegazi, Y. Al-Dossari, A. Al-Yahy, A. Al-Sumari, and A. Hilal, “Preprocessing Arabic text on social media,” Heliyon, vol. 7, no. 2, 2021.
[2] M. K. Saad, “The impact of text preprocessing and term weighting on arabic text classification,” Gaza Comput. Eng. Islam. Univ., 2010.
[3] A. Farghaly and K. Shaalan, “Arabic natural language processing: Challenges and solutions,” ACM Trans. Asian Lang. Inf. Process., vol. 8, no. 4, pp. 1–22, 2009.
[4] S. Elkateb et al., “Arabic WordNet and the challenges of Arabic,” in Proceedings of the International Conference on the Challenge of Arabic for NLP/MT, 2006, pp. 15–24.
[5] I. Issa, “Morphological complexity in Arabic spelling and its implication for cognitive processing,” J. Psycholinguist. Res., vol. 52, no. 1, pp. 331–357, 2023.
[6] N. Y. Habash, Introduction to Arabic natural language processing. Springer Nature, 2022.
[7] Y. Samih, “Dialectal Arabic processing using deep learning.” Dissertation, Düsseldorf, Heinrich-Heine-Universität, 2017, 2017.
[8] F. Husain and O. Uzuner, “Investigating the effect of preprocessing arabic text on offensive language and hate speech detection,” Trans. Asian Low-Resource Lang. Inf. Process., vol. 21, no. 4, pp. 1–20, 2022.
[9] A. A. Nafea, M. S. Ibrahim, A. A. Mukhlif, M. M. AL-Ani, and N. Omar, “An Ensemble Model for Detection of Adverse Drug Reactions,” ARO-THE Sci. J. KOYA Univ., vol. 12, no. 1, pp. 41–47, 2024.
[10] Z. Alyafeai, M. S. Al-shaibani, M. Ghaleb, and I. Ahmad, “Evaluating various tokenizers for Arabic text classification,” Neural Process. Lett., vol. 55, no. 3, pp. 2911–2933, 2023.
[11] A. A. Al Shamsi and S. Abdallah, “Text mining techniques for sentiment analysis of Arabic dialects: Literature review,” Adv. Sci. Technol. Eng. Syst. J, vol. 6, no. 1, pp. 1012–1023, 2021.
[12] M. S. S. Sawalha, Open-source resources and standards for Arabic word structure analysis: Fine grained morphological analysis of Arabic text corpora. University of Leeds, 2011.
[13] J. Zakraoui, M. Saleh, S. Al-Maadeed, and J. M. Alja’am, “Arabic machine translation: A survey with challenges and future directions,” IEEE Access, vol. 9, pp. 161445–161468, 2021.
[14] M. Hijjawi and Y. Elsheikh, “Arabic language challenges in text based conversational agents compared to the English language,” AIRCC’s Int. J. Comput. Sci. Inf. Technol., pp. 1–13, 2015.
[15] M. Sawalha, E. Atwell, and M. A. M. Abushariah, “SALMA: standard Arabic language morphological analysis,” in 2013 1st International Conference on Communications, Signal Processing, and their Applications (ICCSPA), 2013, pp. 1–6.
[16] R. Elbarougy, G. Behery, and A. El Khatib, “A proposed natural language processing preprocessing procedures for enhancing arabic text summarization,” Recent Adv. NLP Case Arab. Lang., pp. 39–57, 2020.
[17] M. Alruily, “Classification of arabic tweets: A review,” Electronics, vol. 10, no. 10, p. 1143, 2021.
[18] Y. Matrane, F. Benabbou, and N. Sael, “A systematic literature review of Arabic dialect sentiment analysis,” J. King Saud Univ. Inf. Sci., p. 101570, 2023.
[19] O. Oueslati, E. Cambria, M. Ben HajHmida, and H. Ounelli, “A review of sentiment analysis research in Arabic language,” Futur. Gener. Comput. Syst., vol. 112, pp. 408–430, 2020.
[20] A. Elnagar, S. M. Yagi, A. B. Nassif, I. Shahin, and S. A. Salloum, “Systematic literature review of dialectal Arabic: identification and detection,” IEEE Access, vol. 9, pp. 31010–31042, 2021.
[21] K. Darwish et al., “A panoramic survey of natural language processing in the Arab world,” Commun. ACM, vol. 64, no. 4, pp. 72–81, 2021.
[22] C. P. Chai, “Comparison of text preprocessing methods,” Nat. Lang. Eng., vol. 29, no. 3, pp. 509–553, 2023.
[23] R. Zarnoufi, W. Bachri, H. Jaafar, and M. Abik, “MANorm: A normalization dictionary for Moroccan Arabic dialect written in Latin script,” arXiv Prepr. arXiv2206.09167, 2022.
[24] A. Shazal, A. Usman, and N. Habash, “A unified model for Arabizi detection and transliteration using sequence-to-sequence models,” in Proceedings of the fifth arabic natural language processing workshop, 2020, pp. 167–177.
[25] T. Sabri, O. El Beggar, and M. Kissi, “Comparative study of Arabic text classification using feature vectorization

methods," Procedia Comput. Sci., vol. 198, pp. 269–275, 2022.
[26] K. Darwish, A. Abdelali, H. Mubarak, and M. Eldesouki, "Arabic diacritic recovery using a feature-rich bilstm model," Trans. Asian Low-Resource Lang. Inf. Process., vol. 20, no. 2, pp. 1–18, 2021.
[27] A. Al-Hmouz, G. Latif, J. Alghazo, and R. Al-Hmouz, "Enhanced numeral recognition for handwritten multi-language numerals using fuzzy set-based decision mechanism," Int. J. Mach. Learn. Comput., vol. 10, no. 1, pp. 99–107, 2020.
[28] A. Mekki, I. Zribi, M. Ellouze, and L. H. Belguith, "Tokenization of Tunisian Arabic: a comparison between three Machine Learning models," ACM Trans. Asian Low-Resource Lang. Inf. Process., vol. 22, no. 7, pp. 1–19, 2023.
[29] A. Neme, "An arabic language resource for computational morphology based on the semitic model." Université Paris-Est, 2020.
[30] M. T. Alrefaie, N. E. Morsy, and N. Samir, "Exploring Tokenization Strategies and Vocabulary Sizes for Enhanced Arabic Language Models," arXiv Prepr. arXiv2403.11130, 2024.
[31] M. O. Alhawarat, H. Abdeljaber, and A. Hilal, "Effect of stemming on text similarity for Arabic language at sentence level," PeerJ Comput. Sci., vol. 7, p. e530, 2021.
[32] A. A. Nafea, "Artificial Neural Network and Latent Semantic Analysis for Adverse Drug Reaction Detection Abstract :," J Comput Sci., no. May, 2023, doi: 10.21123/bsj.2023.7988.
[33] H. A. Almuzaini and A. M. Azmi, "Impact of stemming and word embedding on deep learning-based Arabic text categorization," IEEE Access, vol. 8, pp. 127913–127928, 2020.
[34] E. Al-Shammari and J. Lin, "A novel Arabic lemmatization algorithm," in Proceedings of the second workshop on Analytics for noisy unstructured text data, 2008, pp. 113–118.
[35] I. Zeroual and A. Lakhouaja, "Arabic information retrieval: Stemming or lemmatization?," in 2017 Intelligent Systems and Computer Vision (ISCV), 2017, pp. 1–6.
[36] A. Y. Muaad et al., "Arabic document classification: performance investigation of preprocessing and representation techniques," Math. Probl. Eng., vol. 2022, pp. 1–16, 2022.
[37] L. Hickman, S. Thapa, L. Tay, M. Cao, and P. Srinivasan, "Text preprocessing for text mining in organizational research: Review and recommendations," Organ. Res. Methods, vol. 25, no. 1, pp. 114–146, 2022.
[38] M. Maghfour and A. Elouardighi, "Standard and dialectal Arabic text classification for sentiment analysis," in Model and Data Engineering: 8th International Conference, MEDI 2018, Marrakesh, Morocco, October 24–26, 2018, Proceedings 8, 2018, pp. 282–291.
[39] M. E. M. Abo, R. G. Raj, and A. Qazi, "A review on Arabic sentiment analysis: state-of-the-art, taxonomy and open research challenges," IEEE Access, vol. 7, pp. 162008–162024, 2019.
[40] B. A. Ben Ali, S. Mihi, I. El Bazi, and N. Laachfoubi, "A Recent Survey of Arabic Named Entity Recognition on Social Media.," Rev. d'Intelligence Artif., vol. 34, no. 2, pp. 125–135, 2020.
[41] W. Salloum and N. Habash, "Unsupervised Arabic dialect segmentation for machine translation," Nat. Lang. Eng., vol. 28, no. 2, pp. 223–248, 2022.
[42] C. Chola and A. Y. Muaad, "Investigating the Impact of Preprocessing Techniques and Representation Models on Arabic Text Classification Using Machine Learning."
[43] M. Khairy, T. M. Mahmoud, A. Omar, and T. Abd El-Hafeez, "Comparative performance of ensemble machine learning for Arabic cyberbullying and offensive language detection," Lang. Resour. Eval., pp. 1–18, 2023.
[44] M. Nesca, A. Katz, C. K. Leung, and L. M. Lix, "A scoping review of preprocessing methods for unstructured text data to assess data quality," Int. J. Popul. Data Sci., vol. 7, no. 1, 2022.
[45] M. Chua, D. Van Esch, N. Coccaro, E. Cho, S. Bhandari, and L. Jia, "Text normalization infrastructure that scales to hundreds of language varieties," in Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018), 2018.
[46] H. Mulki, H. Haddad, M. Gridach, and I. Babaoglu, "Empirical evaluation of leveraging named entities for Arabic sentiment analysis," arXiv Prepr. arXiv1904.10195, 2019.
[47] D. Said, N. M. Wanas, N. M. Darwish, and N. Hegazy, "A study of text preprocessing tools for arabic text categorization," in The second international conference on Arabic language, 2009, pp. 230–236.
[48] I. Guellil, H. Saâdane, F. Azouaou, B. Gueni, and D. Nouvel, "Arabic natural language processing: An overview," J. King Saud Univ. Inf. Sci., vol. 33, no. 5, pp. 497–507, 2021.
[49] P. Przybyła and A. J. Soto, "When classification accuracy is not enough: Explaining news credibility assessment," Inf. Process. Manag., vol. 58, no. 5, p. 102653, 2021.
[50] M. Oudah, A. Almahairi, and N. Habash, "The impact of preprocessing on Arabic-English statistical and neural machine translation," arXiv Prepr. arXiv1906.11751, 2019.
[51] D. Gamal, M. Alfonse, S. M. Jiménez-Zafra, and M. Aref, "Case Study of Improving English-Arabic Translation Using the Transformer Model.," Int. J. Intell. Comput. Inf. Sci., vol. 23, no. 2, pp. 105–115, 2023.

[52] H. Alamoudi et al., “Arabic Sentiment Analysis for Student Evaluation using Machine Learning and the AraBERT Transformer,” Eng. Technol. Appl. Sci. Res., vol. 13, no. 5, pp. 11945–11952, 2023.

[53] S. M. Sherif et al., “Lexicon annotation in sentiment analysis for dialectal Arabic: Systematic review of current trends and future directions,” Inf. Process. Manag., vol. 60, no. 5, p. 103449, 2023.

[54] A. Waheed, B. Talafha, P. Sullivan, A. Elmadany, and M. Abdul-Mageed, “VoxArabica: A Robust Dialect-Aware Arabic Speech Recognition System,” arXiv Prepr. arXiv2310.11069, 2023.