Sentiment Analysis of Libyan Dialect

FDS Final Report
Abdullah Habberrih 2184602, Almahdi Elzalouk 2184604, Sebhat Adamu 1999074
Sapienza University

Abstract— **The popularity of social media has coincided with the need for social media sentiment analysis (SA) to gauge people's opinions on several issues. Nevertheless, although Arabic language is known to have a relatively large number of speakers, it is found that most of studies target Modern Standard Arabic (MSA), while Dialectal Arabic (DA), like Libyan Arabic, remains largely ignored. This study seeks to determine the extent to which machine learning (ML) algorithms can adequately analyze sentiments of poetry written specifically in Libyan dialects. For this purpose, the corpus of Libyan poems which were classified as either positive or negative was used to test the effectiveness of three ML models: SVM, NB and LR. It was predicted that all data would require pre-processing such as tokenization, normalization, stemming, lemmatization, stopword removal etc. The TF-IDF with N-grams was employed in feature extraction. In total, six experiments were applied with an aim of determining the optimal parameters set for feature extraction and preprocessing. Among the classifiers SVM was found to be the best achieving of the three 74.63% when lemmatization with Weighted Unigrams and Unigrams in combination with features.**

Keywords—Sentiment Analysis, Arabic Language, Libyan Dialect, Machine Learning

I. Introduction

The surge in social media usage as a platform for individuals to express their opinions on various products and topics [1] has created a demand for the analysis of textual data. To meet this demand, many researchers have employed Natural Language Processing (NLP) methods to investigate people's attitudes and opinions [2]. One particularly useful NLP method is Sentiment Analysis (SA). SA is a task that involves extracting the sentiment from a given text, which can be classified into three categories: positive, negative, and neutral. Some classification schemes also include strongly negative and strongly positive sentiments [1].

As noted by sources [3, 5], the Arabic language is presently ranked as the fifth most widely spoken language in the world, boasting over 350 million native speakers. Within the Arabic language, there exist three distinct variations, namely Classical Arabic (CA), Modern Standard Arabic (MSA), and Dialectal Arabic (DA) [2]. CA, the language utilized in writing the Quran, is considered a classical form of Arabic. MSA, on the other hand, is commonly employed in political, journalistic, literary, and educational contexts. DA is an informal variant of Arabic that is used in daily communication, and it varies from country to country and even from city to city.

The Libyan dialect has limited research articles on sentiment analysis compared to other dialects, such as Saudi [13, 14, 15], Tunisian [16, 17, 18, 19], and Algerian [19, 20, 21, 22]. This study focuses on exploring the effectiveness of three ML techniques in processing Libyan Arabic dialects, which are: Support Vector Machine (SVM), Naive Bayes (NB), and Logistic Regression (LR). It emphasizes evaluating various preprocessing strategies and feature extraction methods, analyzing the results for sentiment classification, and comparing the findings with a previous study that utilized the same dataset.

II. Literature Reviews

Recent studies have explored sentiment analysis across various Arabic dialects using machine learning (ML) and deep learning (DL) techniques. In 2021, Abugharsa [1] compared ML methods and a DL-based tool, Mazajak, on Misurata sub-dialect poems. ML classifiers, including Logistic Regression, Random Forest, Naïve Bayes, and Support Vector Machine (SVM), outperformed the Mazajak tool (based on a CNN algorithm for Modern Standard Arabic), achieving 68.0% accuracy compared to Mazajak's 60.66%. In 2022, Omar et al. [5] analyzed customer opinions of Libyan telecommunication companies using Twitter data annotated as positive or negative. Through extensive preprocessing and TF-IDF feature extraction, the study tested five classifiers (SVM, LR, NB, KNN, and DT) across three experiments. SVM achieved the highest accuracy (80.67%) for Libyana, NB excelled for Almadar Aljadid (81.19%), and DT performed best for Libya Telecom and Technology (75%). Moreover, in 2023, Errami et al. [6] investigated sentiment analysis of Moroccan dialect using ML and DL approaches on a Twitter dataset labeled as positive, negative, and neutral. The study applied preprocessing techniques such as cleaning, tokenization, and stemming, with feature extraction using TF-IDF, N-grams, and word embeddings. Among the classifiers tested, SVM achieved the highest accuracy (68.59%), outperforming other ML methods and the LSTM-based DL model. Across these studies, ML methods consistently demonstrated superior performance compared to DL approaches for sentiment analysis in Arabic dialects.

Building on these studies, this study focuses on sentiment analysis of the Libyan dialect, particularly in poetry, using ML techniques. It evaluates the impact of various preprocessing strategies and feature extraction methods, applying classification algorithms to determine their effectiveness. Additionally, it compares the findings with previous study that used the same dataset to provide a comprehensive analysis of sentiment classification performance.

III. Methodology

The methodology of this study is presented in Figure 1.

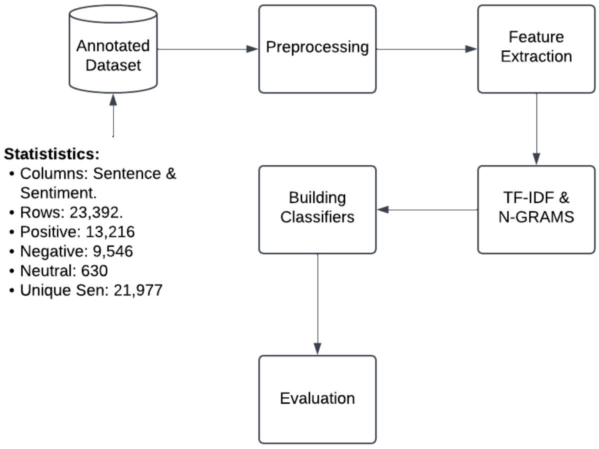

Figure 1. The Methodology of Study

The dataset used in this study consists of Libyan sentences annotated from [1]. This particular dataset focuses on poetry in the Libyan dialect and comprises 23,392 records that have been labeled as either positive, negative, or neutral. Table 1 illustrates the distribution of the dataset samples across each class.

Tabel 1. Number of Samples in the Dataset

Class	Number of Samples
Positive	13,216
Negative	9,546
Neutral	630
Total	23,392

The dataset contained columns that were beyond the scope of this study, such as 'Metaphor', 'Keywords', and 'Source Domain'. In order to align the dataset with the objectives of this study, the irrelevant columns were removed, along with the neutral class and the duplicated rows. Furthermore, Table 2 highlights some of the samples of the positive and the negative classes.

Table 2. Some Examples of Dataset

Sentiment	Sentence	Meaning in English
Positive	ضحكت ضحك قلبي و رفرف طاير	She laughed—a laughter that made my heart laugh and flutter like a bird in flight.
Negative	و عارف و اني عارف كلها تغبى	I know, indeed I know, she only pretends not to understand

The preprocessing phase plays a crucial role in sentiment analysis (SA) as it enhances the quality of data and improves the overall performance of the analysis. Tokenization involves segmenting the text stream into individual words or tokens using a delimiter, such as whitespace or punctuation characters. Sentences were tokenized into individual words using CAMeL Tools. Normalization transforms text to achieve consistency and standardize its form [7]. It includes the removal of diacritics, normalization of Alef variants (أ, إ, آ) to ا, conversion of Teh Marbuta (ة) to Heh (ه), and normalization of Alef Maksura (ى) to Ya (ي). The process of removing stopwords eliminates natural language words that carry little meaning [7], achieved here using a predefined stopword list. Stemming involves light and root stemming, aiming to reduce words to their uninflected base forms. While stems may differ from roots, stemming is useful as related words often map to the same stem, even if the stem is not a valid root [7]. Lemmatization, a linguistic method, analyzes a word's morphology and removes its inflectional suffix to generate its base form or lemma, corresponding to its dictionary entry [8]. In this step, ISRI Stemmer and Qalsadi Lemmatizer were applied based on the experiments.

After preprocessing the data, the next step is feature selection and extraction, which is crucial for identifying the most relevant features for sentiment analysis while removing irrelevant, redundant, and noisy data [9]. In the context of opinion mining, features are words, terms, or phrases that strongly express either positive or negative sentiments, significantly influencing the sentiment orientation of the text [10]. Two main feature extraction techniques are utilized in this process: TF-IDF and N-grams. TF-IDF combines two components: Term Frequency (TF), which measures how often a word appears in a document, and Inverse Document Frequency (IDF), which reduces the impact of commonly occurring words by highlighting less frequent but significant terms [11]. N-grams, on the other hand, generate combinations of adjacent words or letters of length NNN, such as Unigrams (N=1), Bigrams (N=2), and Trigrams (N=3) [12]. Unigrams focus on single words, while Bigrams and Trigrams consider two- and three-word sequences, respectively. Combining these techniques results in a broader set of features, offering a more comprehensive representation of the dataset.

IV. Experiments and Results

Six experiments were conducted in this study using three ML classifiers, focusing on various preprocessing techniques and feature extraction methods. The preprocessing techniques explored in the experiments include Stemming, Lemmatization, and Stop-word Removal, while Tokenization, Normalization, and Data Cleaning were consistently applied across all experiments. The feature extraction methods included TF-IDF and different combinations of N-grams: Unigrams, Bigrams, Trigrams, Unigrams with Bigrams, Unigrams with Trigrams, and Bigrams with Trigrams. The experiments were as follows:

1. Stemming with Stop-word Removal.
2. Stemming without Stop-word Removal.
3. Lemmatization with Stop-word Removal.

4. Lemmatization without Stop-word Removal.
5. Stemming and Lemmatization with Stop-word Removal.
6. Stemming and Lemmatization without Stop-word Removal.

The purpose of these experiments was to determine the optimal preprocessing and feature extraction techniques for each classifier. The results of these experiments are presented in Table 3.

Table 3. Experiments Results

Experiment	Classifier	N-gram	Accuracy (%)
Stemming with Stopwords Removal	SVM	(1, 3)	70.7665
	Naive Bayes	(1, 2)	70.0637
	Logistic Regression	(1, 2)	70.0417
Stemming without Stopwords Removal	SVM	(1, 3)	73.7755
	Naive Bayes	(1, 2)	71.6231
	Logistic Regression	(1, 2)	72.1502
Lemmatization with Stopwords Removal	SVM	(1, 2)	71.7769
	Naive Bayes	(1, 1)	70.8105
	Logistic Regression	(1, 1)	70.6347
Lemmatization without Stopwords Removal	SVM	(1, 3)	74.6321
	Naive Bayes	(1, 2)	72.282
	Logistic Regression	(1, 2)	72.26
Stemming and Lemmatization with Stopwords Removal	SVM	(1, 2)	71.2717
	Naive Bayes	(1, 2)	70.459
	Logistic Regression	(1, 2)	70.2614
Stemming and Lemmatization without Stopwords Removal	SVM	(1, 3)	74.0611
	Naive Bayes	(1, 2)	71.9526
	Logistic Regression	(1, 2)	72.26

The study focuses on evaluating different classifiers and preprocessing techniques for sentiment analysis of Libyan dialect poems. Preprocessing methods such as stemming, lemmatization, and stopword removal, along with the selection of classifiers and N-gram ranges, had a significant impact on the performance of the models. SVM achieved the highest accuracy of 74.63% when stemming was used, stopwords were not removed, and a combination of Unigrams and Trigrams was applied. This shows that SVM is effective at identifying complex patterns in the data when paired with well-chosen N-grams. The effect of stopword removal varied depending on the combination of techniques used. Naïve Bayes also performed well but did not surpass SVM, achieving 72.28% accuracy with lemmatization and no stopword removal. Logistic Regression showed consistent performance, with its highest accuracy being 72.26%, which was slightly lower than the other two classifiers.

The combination of Unigrams and Trigrams worked best for SVM because it captures both single words and meaningful word sequences. On the other hand, simpler N-gram ranges, such as Unigrams & Bigrams, worked well for Naïve Bayes and Logistic Regression.

Using both stemming and lemmatization together gave competitive results. SVM reached 74.06% accuracy when stopwords were not removed, highlighting the importance of carefully selecting preprocessing techniques and feature extraction methods for each classifier to achieve the best results. Figure 2 shows the results.

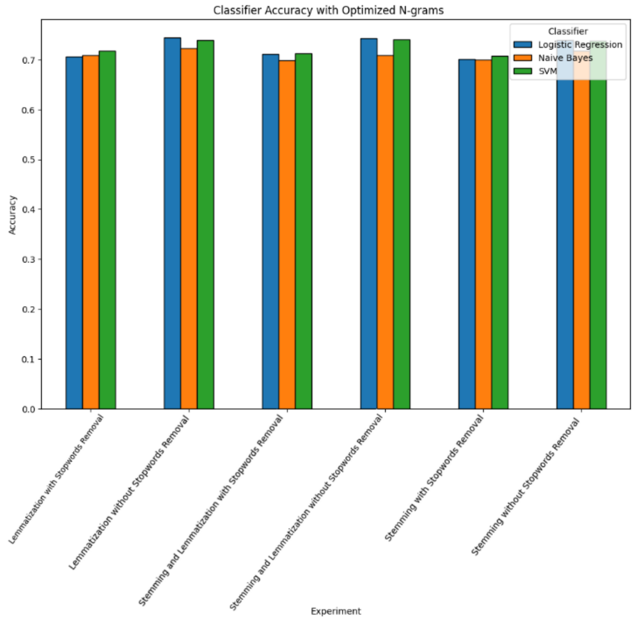

Figure 2. Experiments Results

This study demonstrates a significant improvement compared to the findings from [1], which analyzed sentiment in Libyan dialect poems using machine learning (ML) and deep learning (DL) techniques. In [1], Naïve Bayes was the best ML classifier, achieving an accuracy of 69.0%, while the deep learning model Mazajak, based on a CNN, reached only 60.66%.

In this study, a higher accuracy of 74.63% was achieved using the SVM classifier with a combination of Unigrams and Trigrams. This improvement emphasizes the importance of optimizing feature selection, as including N-grams such as Unigrams, Bigrams, and Trigrams helped enhance SVM performance. Unlike the approach in [1], where N-grams were not explicitly optimized, this study highlights how careful feature extraction can significantly improve sentiment analysis results.

Another notable difference is the choice of classifiers. While [1] identified Naïve Bayes as the best-performing model, the results here show that SVM outperforms it. This improvement is likely due to better preprocessing techniques, such as stopword removal, stemming, and normalization. These steps reduced noise in the data, enabling SVM to learn patterns more effectively and achieve higher accuracy in sentiment analysis.

V. Conclusion and Future Work

Sentiment analysis is a crucial application of the Natural Language Processing (NLP) field that helps understand public sentiment regarding a product, service, place, or even a public issue. However, developing NLP models and tools for Arabic dialects is challenging due to limited written resources for many of these dialects. This study highlights the effectiveness of machine learning techniques in sentiment analysis of Libyan dialect poems. The results demonstrate the impact of preprocessing and feature extraction methods on classifier performance, with SVM achieving the best accuracy. The comparison with previous studies also shows the importance of optimizing these techniques to improve sentiment analysis results. Future work could explore additional deep learning approaches and hybrid models to further enhance performance.

Division of Work

The study tasks were divided as follows: Almahdi with the help of Abdullah handled data preprocessing, including cleaning, tokenization, normalization, stopword removal, stemming, and lemmatization using tools like CAMeL Tools and Qalsadi Lemmatizer. Feature extraction methods, such as TF-IDF and N-grams (Unigrams, Bigrams, and Trigrams), were applied to prepare the dataset. Also, with the help of Abdullah, Subhat focused on training the classifiers (SVM, Naïve Bayes, and Logistic Regression) using scikit-learn. We conducted six experiments with different preprocessing techniques and N-gram combinations and evaluated the models' performance using metrics like accuracy. Abdullah analyzed the results, highlighting that SVM achieved the best accuracy (74.63%) with Lemmatization and Unigrams and Trigrams. He also prepared the final report, comparing the findings with previous studies and suggesting future work.

References

[1] A. Abugharsa, “Sentiment Analysis in Poems in Misurata Sub-dialect: A Sentiment Detection in an Arabic Sub-dialect”, IJCT, vol. 21, pp. 103–114, Sep. 2021.

[2] S. Alyami, A. Alhothali, and A. Jamal, “Systematic literature review of arabic aspect-based sentiment analysis,” J. King Saud Univ. Inf. Sci., vol. 34, no. 9, pp. 6524–6551, 2022.

[3] A. O. O. Alshutayri and E. Atwell, “Exploring Twitter as a source of an Arabic dialect corpus,” Int. J. Comput. Linguist., vol. 8, no. 2, pp. 37–44, 2017.

[4] I. Guellil and F. Azouaou, “Arabic dialect identification with an unsupervised learning (based on a lexicon). application case: Algerian dialect,” in 2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES), 2016, pp. 724–731. Challenges in Sentiment Analysis for Arabic Social Networks.

[5] A. Omar, M. Essgaer, and K. M. S. Ahmed, “Using Machine Learning Model To Predict Libyan Telecom Company Customer Satisfaction,” in 2022 International Conference on Engineering & MIS (ICEMIS), IEEE, 2022, pp. 1–6.

[6] M. Errami, M. A. Ouassil, R. Rachidi, B. Cherradi, S. Hamida, and A. Raihani, “Sentiment Analysis on Moroccan Dialect based on ML and Social Media Content Detection,” Int. J. Adv. Comput. Sci. Appl., vol. 14, no. 3, 2023.

[7] A. Shoukry and A. Rafea, “Preprocessing Egyptian Dialect Tweets for Sentiment Mining,” in The Fourth Workshop on Computational …, 2012, pp. 47–56.

[8] S. Symeonidis, D. Effrosynidis, and A. Arampatzis, “A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis,” Expert Syst. Appl., vol. 110, pp. 298–310, 2018.

[9] A. A. Sayed, E. Elgeldawi, A. M. Zaki, and A. R. Galal, “Sentiment analysis for arabic reviews using machine learning classification algorithms,” in 2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE), IEEE, 2020, pp. 56–63.

[10] E. Haddi, X. Liu, and Y. Shi, “The role of text pre-processing in sentiment analysis,” Procedia Comput. Sci., vol. 17, pp. 26–32, 2013.

[11] A. A. Al Shamsi and S. Abdallah, “Sentiment Analysis of Emirati Dialect,” Big Data Cogn. Comput., vol. 6, no. 2, p. 57, 2022.

[12] W. B. Cavnar and J. M. Trenkle, “N-gram-based text categorization,” in Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval, Las Vegas, NV, 1994, p. 14.

[13] G. Alwakid, T. Osman, and T. Hughes-Roberts, “Challenges in sentiment analysis for Arabic social networks,” Procedia Comput. Sci., vol. 117, pp. 89–100, 2017.

[14] A. Al-Thubaity, Q. Alqahtani, and A. Aljandal, “Sentiment lexicon for sentiment analysis of Saudi dialect tweets,” Procedia Comput. Sci., vol. 142, pp. 301–307, 2018.

[15] L. Almuqren and A. Cristea, “AraCust: a Saudi Telecom Tweets corpus for sentiment analysis,” PeerJ Comput. Sci., vol. 7, p. e510, 2021.

[16] A. Mekki, I. Zribi, M. Ellouze, and L. H. Belguith, “A Tunisian benchmark social media data set for COVID-19 sentiment analysis and sarcasm detection,” 2022.

[17] S. Mdhaffar, F. Bougares, Y. Esteve, and L. Hadrich-Belguith, “Sentiment analysis of tunisian dialects: Linguistic ressources and experiments,” in Third Arabic Natural Language Processing Workshop (WANLP), 2017, pp. 55–61.

[18] S. Jaballi, S. Zrigui, M. A. Sghaier, D. Berchech, and M. Zrigui, “Sentiment Analysis of Tunisian Users on Social Networks: Overcoming the Challenge of Multilingual Comments in the Tunisian Dialect,” in International Conference on Computational Collective Intelligence, Springer, 2022, pp. 176–192.

[19] K. Abidi and K. Smaïli, “Creating multi-scripts sentiment analysis lexicons for Algerian, Moroccan and Tunisian dialects,” in 7th International Conference on Data Mining (DTMN 2021) Computer Science Conference Proceedings in Computer Science & Information Technology (CS & IT), 2021.

[20] B. Klouche, S. M. Benslimane, and N. Mahammed, “Sentiment analysis of Algerian dialect using a deep learning approach,” in International Conference on Artificial Intelligence and its Applications, Springer, 2021, pp. 122–131.

[21] I. Guellil, A. Adeel, F. Azouaou, and A. Hussain, “Sentialg: Automated corpus annotation for algerian sentiment analysis,” in Advances in Brain Inspired Cognitive Systems: 9th International Conference, BICS 2018, Xi’an, China, July 7-8, 2018, Proceedings 9, Springer, 2018, pp. 557–567.

[22] A. Abdelli, F. Guerrouf, O. Tibermacine, and B. Abdelli, “Sentiment analysis of Arabic Algerian dialect using a supervised method,” in 2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS), IEEE, 2019, pp. 1–6.