Sentiment Analysis of Arabic Reviews of Restaurants Using Pretrained Language Models

Mena Allah Hassaneen[1*], Samer Ibrahim [2], Sara Hamdy[3], Amr A.Awamry[4]

[1*]Electrical Department , Benha Faculty of Engineering, Benha, Egypt.
[2]Computer Engineering department, October University for Modern Sciences and Arts (MSA), cairo, Egypt.
[3]Electrical Department , Benha Faculty of Engineering, Benha, Egypt.
[4]Electrical Department , Benha Faculty of Engineering, Benha, Egypt.

*Corresponding author(s). E-mail(s): m.mohammed57015@beng.bu.edu.eg;
Contributing authors: saibrahim@msa.edu.eg; sara.hamdy@bhit.bu.edu.eg; amr.awamry@bhit.bu.edu.eg;

Abstract

Sentiment analysis, a core task within the swiftly advancing field of Natural Language Processing (NLP), seeks to discern the emotional tone behind written words. Think about online restaurant reviews: every day, tons are created as people share their dining experiences, potentially helping others choose where to eat. Yet, sifting through this enormous data pile poses a significant challenge; as such, applying natural language processing methods becomes essential to extract useful insights Dwivedi et al. (2023). The availability of Arabic NLP tools and studies is often less than what exists for English, a situation caused by the Arabic language's structural complexity, cultural context, and ambiguity, which highlights the importance of future studies. Most previous work has utilized either classical machine learning or traditional deep learning. In this work, I leverage various techniques for preprocessing Arabic text to boost model effectiveness. I then fine-tune various pre-trained transformer-based models like XLMRoBERTa, two AraBERT versions, CAMELBERT, MARBERT, QARIB, ArabicBERT, DISTILBERT, GigaBERT, and ARAELECTRA. I also compare model performance according to various metrics, considering different hyperparameter configurations. To that end, I also introduce a new large binary dataset consisting of 52,493 Arabic Google Maps reviews, manually labeled as positive or negative. Evaluating these models involved accuracy, recall, precision, and

F1 score. Ultimately, MARBERT demonstrated the best results, , showing an accuracy of 97.23%, precision of 97.25%, recall of 97.23%, and an F1 score of 97.24%.

Keywords: Arabic text classification, Arabic sentiment analysis, Google maps reviews, pretrained language models, BERT-based models, large scale datasets

1 Introduction

Google Maps, a service offered via the web by Google, is now a go-to resource when people want to find information about places. People can share what they think of spots, like restaurants, mentioning things like food, prices, and how good the service is. These user reviews really matter, helping others decide where to eat and also giving restaurant owners feedback Dwivedi et al. (2021). Reviews really have an effect on what people choose, influencing potential customers with other diners' stories. This ongoing interaction between users and businesses shows how digital spaces impact consumer choices and shape competition in the restaurant world. In today's competitive climate, making sure customer service is high-quality, and that customers see those services as valuable, is critical. A major challenge is dealing with tons of data, especially in Arabic. This difficulty arises because Arabic has complex word structures, lots of ambiguity, many different dialects, grammar and spelling errors, and often uses informal language in reviews with idioms and slang, plus a shortage of Arabic resources. Traditional machine learning struggles to grasp the context or meaning of Arabic text, creating a need for better, more specialized models. Early deep learning NLP models, like bag of words Qader et al. (2019), n-grams Brown et al. (1992) , and term frequency inverse document frequency (TF-IDF) Das et al. (2023) , showed improvements over older methods. Then, models like word2vec Mikolov et al. (2013) and Glove Pennington et al. (2014) offered better ways to represent sentences. Still, understanding sentences in context remained a problem. (Attention Is All You Need) Vaswani et al. (2023) transformer model changed things, greatly improving our grasp of text context and leading to advancements in fine-tuning for various tasks. Models like AraBERT Antoun et al. (2020), XLMRoBERTa Conneau et al. (2020), CAMELBERT Inoue et al. (2021), and MARBERT Abdul-Mageed et al. (2021), trained on big datasets, have emerged. This study aims to fine-tune these models and do a comparison using Google Maps reviews data. This will address a gap in Arabic text classification, capturing language features, informal writing, and dialect variations to improve classification. The key contributions of this paper are: first, building a 52,493-entry dataset with positive and negative restaurant reviews from Google Maps, annotated manually and with an uneven balance; second, outlining a process for cleaning, preprocessing, and fine-tuning data, along with essential tools and libraries for Arabic text classification. Also, it involves fine-tuning ten models like XLMRoBERTa , two AraBERT versions, CAMELBERT , MARBERT , QARIB Abdelali et al. (2021), ArabicBERT Safaya et al. (2020), DISTILBERT Sanh et al. (2020), GigaBERT Lan et al. (2020), and ARAELECTRA Antoun et al. (2021), on the Arabic dataset. Finally, it assesses model performance using metrics adapted for Arabic text. The paper's structure is as follows: Section 2 covers related work; Section 3 details the methodology; Section 4 explains the experimental setup; Section 5 discusses the limitations; , Section 6 provides conclusions and Section 7 for future research directions.

2 Literature Review

Arabic holds a place among the official languages of the United Nations . This underscores its global importance, suggesting a need for natural language processing (NLP) solutions crafted for its speakers. Over time, the field of Arabic text classification has seen considerable growth. This is clear when we look at the different academic works that have deepened our understanding of this important subject. Many studies have taken a close look at the methods used in Arabic text classification, as well as how these methods are used in real-world situations.

2.1 Introduction to Arabic text classification

Sentiment analysis is an NLP field that looks into people's opinions, feelings, attitudes, and even emotions about a range of things, products, services, organizations, issues, events, and topics Oueslati et al. (2020). This way of framing things shows just how messy and layered sentiment analysis can be. studies show the need for better keyword extraction and classifiers in Arabic text classification Addi and Ezzahir (2024), Ouassil et al. (2024). Recent work by Ghallab et al. (2020) shows that, in most cases, the boom of Arabic content online has pushed this field into the spotlight; there's just so much data now that advanced NLP techniques become essential to sift through it all. They jump right into the issues facing Arabic Sentiment Analysis, noting that a real stumbling block is the persistent lack of dedicated resources and tools. This shortfall not only holds back the creation of accurate sentiment analysis setups but also slows related NLP progress since the scarcity of annotated corpora and specialized lexicons really cripples the overall effort. They suggest latest trends in ASA like giving more attention to text in MSA and DA than MSA only because of enormous presence of DA in daily life applications, especially social media

2.2 Challenges in Arabic

Sentiment analysis and text classification come with a fair share of issues think review quality, fuzzy polarity, spam clutter, and tight domain rules Oueslati et al. (2020). But Arabic throws in extra hurdles that are tougher to crack. Chouikhi et al. (2021) kind of walks us through the mess: Arabic words tend to twist their meaning thanks to a tangle of prefixes, suffixes, and inflections so much so that a single term might carry loads of nuance, making things generally trickier. Then there's the data problem; in most cases, the total lack of big, annotated Arabic corpora really puts a damper on training quality and accuracy, the underutilization of available lexicons Almurqren et al. (2024). And don't even get me started on the dialects. Arabic is awash with regional variants that even differ within the same country, meaning reviews written in local lingo just don't stick to any standard, complicating sentiment generalizations further. All these layered challenges continue to bog down Arabic sentiment analysis.

2.3 Classical machine learning approaches

Musleh et al. (2023) rolled out an interesting model that sorts Arabic YouTube comments into positive or negative buckets. They put together a dataset of 4,212 hand-labeled Arabic comments. They tried out six well-tried machine learning approaches: Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, Naïve Bayes, and Random Forest. Curiously, the Naïve Bayes method came out on top, hitting a 94.62% accuracy rate; this result shows how even classic techniques can handle the subtle nuances of Arabic sentiment

2.4 Feature extraction

Omar et al. (2021) presents a study that delved into multilabel Arabic text classification in a rather unconventional style. They set out to build a vast dataset of about 44,000 posts and tweets using a blend of careful manual annotation mixed with semi-supervised methods. The paper rambles through the challenges posed by Arabic. Instead of sticking to one routine method, they experimented with three ways to represent features: a straightforward Bag of Words (BoW), TF-IDF (that's Term Frequency-Inverse Document Frequency), and a method using N-grams. After a bunch of rigorous tests, they noticed that pairing N-grams with a LinearSVC classifier ended up giving them outstanding results roughly a 97.8% accuracy score.

2.5 Deep learning approaches

Alqarni and Rahman (2023) gathered a wide range of tweets concerning COVID-19, They then turned their attention to the tricky task of preprocessing Arabic text, which is, in most cases, rather challenging given its unique twists. Once these early steps were sorted, the researchers wedded advanced deep learning methods, using methods like CNN and BiLSTM to effectively sift through and classify the COVID-19 tweets. In doing so, they not only highlighted the strengths and the not-so-strong points of their approach but also laid down a sturdy foundation for comparing future methods as Guan and Treude (2024). The blend of CNN with BiLSTM stands as a solid example of how sophisticated text classification has become a sign of the rapid advances in deep learning that are reshaping NLP by Abaimov and Bianchi (2021). All in all, this work contributes to an area that's moving fast, suggesting that deep learning models hold the potential to drastically boost outcomes in tackling text classification challenges Minaee et al. (2021).

2.6 Pretrained language models (transformer-based models)

The rise of the Transformers architecture Vaswani et al. (2023) has really changed Natural Language Processing (NLP), showing impressive results in different tasks. Studies show how to use these models to get great results in text classification and sentiment analysis. For example, AraBERT stands out in spotting toxic Arabic tweets, as

Koshiry et al. (2023) pointed out. Also, Galal et al. (2024) looked at sarcasm in Arabic text, creating the ArSarcasT corpus (26,000 annotated tweets). They tweaked four datasets using AraBERT, MARBERT, and QARIB, doing better at finding sarcastic tweets than before. Furthermore, Mousa et al. (2024) reviewed how text classification for offensive Arabic content has changed in the last ten years. They started with simpler methods like Support Vector Machines (SVM), Naive Bayes (NB), and k-Nearest Neighbors (KNN), then moved to deep learning like Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNN). Eventually, they got to transfer learning models AraBERT, ArabicBERT, GigaBERT, XLMRoBERTa, MBERT Libovický et al. (2019), and QARIB, which show progress but also highlight the limits of old models with hard data and the need for better accuracy, as Alammary (2022) said. Abu Farha and Magdy (2021) compared some new Arabic language models for sentiment analysis and sarcasm detection using the ArSarcasm-v2 dataset.Chowdhury et al. (2020) investigated the efficiency of using transformer-based models in tasks of text classification of the formal and formal short texts and the importance of training models on dataset mix of formal and informal text to achieve higher performance compared to models trained on dataset of formal text only.

2.7 user-generated content analysis

Analyzing Google Maps reviews presents both significant importance and considerable challenges, as demonstrated by Shin et al. (2022), who utilized data crawling, preprocessing, text vectorization, and machine learning in their investigation of Korean-language reviews. This multifaceted strategy is generally critical for a deeper comprehension of user-generated content, and it offers considerable insight into consumer behavior and sentiment analysis, a point also addressed by Chen and Chang (2024). Their research explores Google Maps reviews to evaluate public library service content, highlighting the important role of digital feedback in understanding public library use across six major Taiwanese cities. Akkaya et al. (2024) further support this idea, advocating for the use of user-generated reviews with AI to produce more reliable results by employing advanced digital tools. Their findings show how effectively Google Maps reviews can be used for parks within Istanbul, reinforcing how valuable it is to integrate user-generated content into both academic research and practical applications across different fields.

3 Proposed Method

3.1 Dataset description

Below we will introduce the description of used Dataset :

3.1.1 Dataset collection

This research leverages a dataset compiled from Google Maps user reviews of Egyptian restaurants. To ensure systematic data collection, we utilized Apify, a web scraping tool. It's worth noting that these reviews reflect a broad spectrum of factors; think diverse geographic locations, variations in local dialects, different price ranges, the impact of seasonal changes, and the involvement of varied social classes. Such diversity is key to minimizing bias and ensuring the dataset's representativeness. Indeed, Google Maps reviews are among the most accessible platforms for sharing opinions and ratings, making them a valuable data source for sentiment analysis. Data collection spanned from early 2020 to late 2024. This enables a thorough examination of consumer sentiment trends. Significantly, before this study, no publicly available dataset of Arabic Google Maps reviews specifically tailored for Arabic sentiment analysis was found.

3.1.2 Dataset labeling

To label reviews based on star ratings, several strategies are employed. The primary method relies on the star count itself. Positive reviews are generally those with four or five stars, whereas reviews with one to three stars are considered negative. After this initial step, a careful manual review of the annotations takes place. Revisions and re-annotations are performed as deemed necessary. This iterative process is quite important, particularly as initial analysis showed many errors where star ratings did not match the review content. Typically, the positive class includes reviews recommending restaurants. On the other hand, the negative class typically contains reviews criticizing or raising concerns about the dining experience. This comprehensive approach helps build a sizable and high-quality dataset, useful for research and real-world application. By maintaining strict annotation protocols, we strive to ensure the dataset's breadth and its contribution to understanding consumer sentiment in the restaurant sector.

3.1.3 Dataset analysis

This study explores the use of a dataset we built ourselves. This dataset centers around Google Maps reviews. These reviews have become a subject of interest in sentiment analysis and natural language processing. The dataset consists of 52,493 Arabic reviews, We use this large number of reviews to try to achieve better performance as more data naturally leading to better performance Omar et al. (2021) , Kasongo and Sun (2020). carefully split into positive and negative classes. To go into more detail, 35,716 reviews were marked as positive, while 16,777 were considered negative. The sentiment classification is important for assessing how well sentiment analysis algorithms handle real data. This improves our understanding of Arabic sentiment on digital platforms. Figure 1 demonstrates the distribution of classes. Length of reviews by letters before cleaning is in range of (max = 2815 letters, min = 1 letters) can be visualized in Figure 2

3.2 Dataset preprocessing

In Natural Language Processing (NLP), preprocessing techniques significantly impact the effectiveness of various tasks, especially in sentiment analysis and text classification where high accuracy is key. Given that our dataset consists of reviews written in informal Arabic dialects by diverse contributors, preprocessing becomes particularly important. These dialects often contain unique linguistic errors, including emojis,

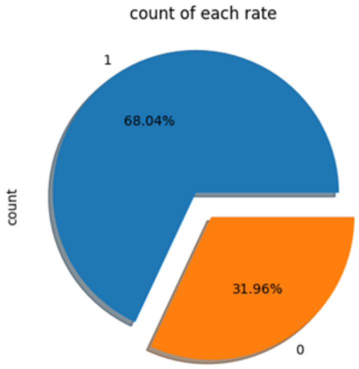


Fig. 1 shows the percentage of classes 1 for Positive Reviews and 0 for Negative Reviews before cleaning

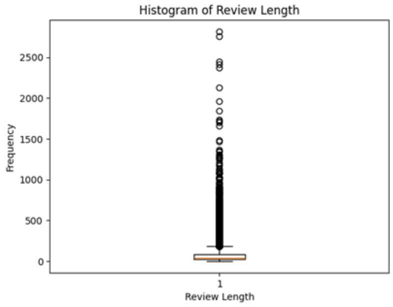
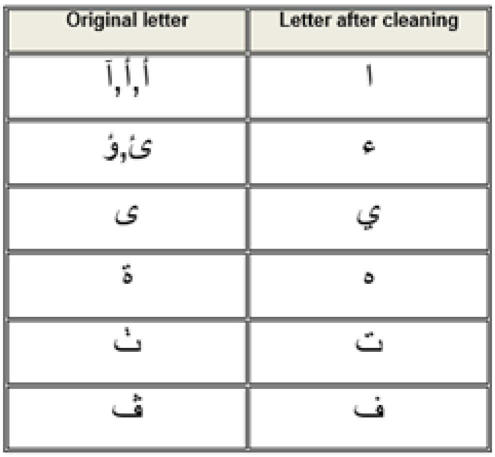


Fig. 2 shows the histogram of reviews Lengths before cleaning

URLs, hashtags, and mention tags, which further amplifies the need for a thorough cleaning of redundant or extraneous text elements. A rigorous preprocessing framework standardizes the input, allowing for a more precise sentiment analysis across different dialects and linguistic styles, and maintains language nuances while improving the data quality for later modeling tasks Abdul-Rahman et al. (2025). This cleaning process not only optimizes the training workflow but also enhances the models' ability to understand the core text leading to improved performance.

3.2.1 Dataset cleaning

This phase is about boosting the text processing accuracy. We start by ditching any empty rows, because those can mess with our analysis. Then, we get rid of those one-letter words, since they don't really add anything meaningful to the data. Also, spotting and removing duplicate reviews is a must, as they might introduce bias into the results. Reviews that are longer than 800 characters are also taken out to keep things consistent and focused, this choice is supported by the fact that most reviews are shorter anyway. This is important because it directly improves accuracy. After that, it's also necessary to eliminate diacritics, to keep the text data uniform. Spelling



Original letter	Letter after cleaning
أ,أ,آ	ا
ئ,ؤ	ء
ى	ي
ة	ه
ٹ	ت
ڤ	ف

Fig. 3 Examples of Arabic character variations with their regular forms

normalization is a critical step, that involves fixing spelling problems by replacing Arabic character variations with their regular forms as in 3 Figure 7: This is done by using regular expressions (re) library, which studies show can really improve text quality Wang et al. (2020). To enhance model efficacy, extraneous elements are refined via regular expressions to diminish out-of-vocabulary instances by removing special characters, non-Arabic characters, punctuation marks, and web addresses. Hashtags and mentions are substituted with standardized terms, and words containing more than two consecutive repeating letters or spaces are corrected . Each emoji, significant in review contexts, is replaced with its Arabic translation utilizing an emoji library; their value for analysis is crucial Hakami et al. (2022). Lemmatization, usually key in text preparation, may be less critical when using pre-trained models, as they already manage word variations efficiently, given their extensive training. Removing stop words is often not particularly useful since large models mean their exclusion won't substantially improve performance. post-cleaning (involving removing duplicates and very long entries) the dataset contains 51,808 reviews: 35,218 positive and 16,590 negative, as visualized in Figure 4. Review lengths prior to cleaning, as shown in Figure 5, span from 2 to 800 letters.

3.2.2 WORDCLOUD

After cleaning word clouds are used to showcase common words in positive and negative feedback to extract topical insights. The wordcloud library is leveraged to visually represent word frequency, with font size correlating to occurrence rates. We show the most commonly used words in positive and negative reviews in Figure 6.

3.3 Tokenization

Given the fact that models don't inherently grasp words, tokenization and padding stand out as key steps. They segment sentences into word and sub-word lists before turning them into unique numerical forms. Fixing the length of these output number

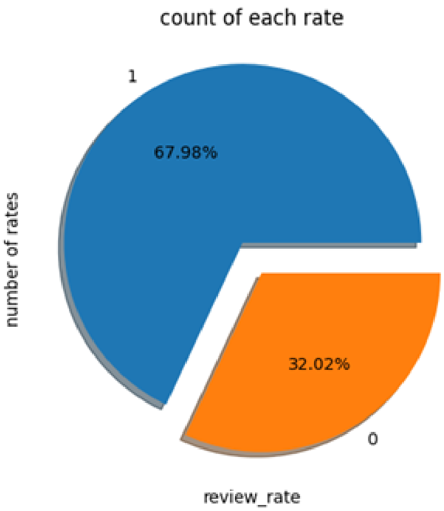


Fig. 4 shows the percentage of classes 2 for Positive Reviews and 0 for Negative Reviews after cleaning

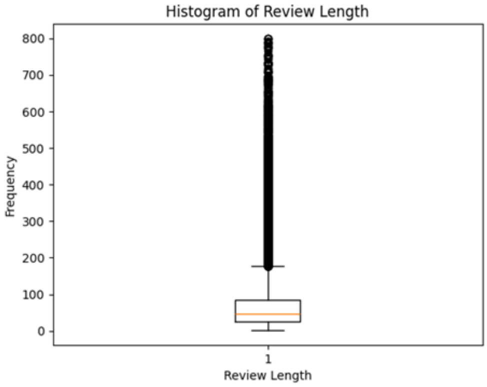


Fig. 5 shows the histogram of reviews Lengths after cleaning

sequences is crucial, even though input texts naturally vary in length. Arabic, a morphologically rich language, encodes significant information within words via prefixes and suffixes. According to Alkaoud and Syed (2020), this poses an embedding learning challenge, due to numerous potential word forms, increasing out-of-vocabulary (OOV) issues, thus growing the model size. In general, transformer-based models need a tokenization step before training. To enable understanding and structured text processing, pre-trained model creators have developed tokenizers converting text into model-compatible formats. Research indicates this structure aids models in achieving greater performance by maintaining word context without pretraining. For our work, we incorporated diverse tokenizers for each pre-trained model. Figure 7 displays the various tokenizer outputs from the sentence

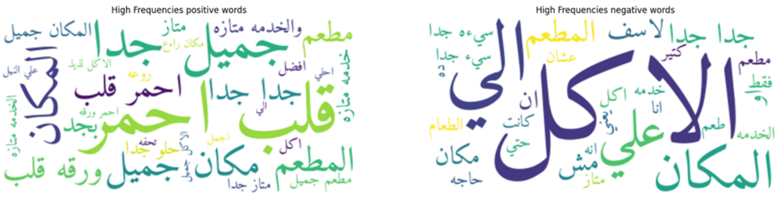
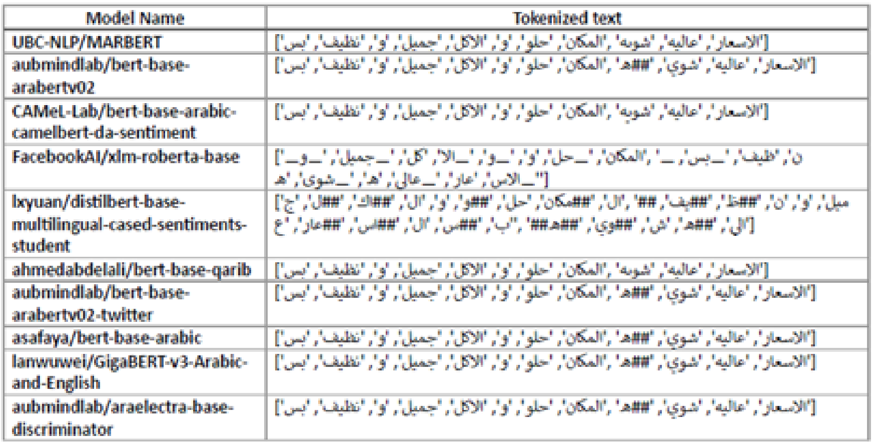


Fig. 6 shows the WordCloud of the Positive and Negative reviews



Model Name	Tokenized text
UBC-NLP/MARBERT	['الاسعار', 'عاليه', 'شويه', 'المكان', 'حلو', 'و', 'الاكل', 'جميل', 'و', 'نظيف', 'بس']
aubmindlab/bert-base-arabertv02	['الاسعار', 'عاليه', 'شوي', '##ه', 'المكان', 'حلو', 'و', 'الاكل', 'جميل', 'و', 'نظيف', 'بس']
CAMeL-Lab/bert-base-arabic-camelbert-da-sentiment	['الاسعار', 'عاليه', 'شويه', 'المكان', 'حلو', 'و', 'الاكل', 'جميل', 'و', 'نظيف', 'بس']
FacebookAI/xlm-roberta-base	['ن', 'ظيف', 'ـبس', 'ـ', 'المكان', 'ـحل', 'و', 'ـو', 'ـالا', 'كل', 'ـجميل', 'ـو_' 'ـالاس', 'عار', 'ـعالي', 'ه', 'ـشوي', 'ه"]
lxyuan/distilbert-base-multilingual-cased-sentiments-student	['ميل', 'و', 'ن', '##ظ', '##يف', '##', 'ال', '##مكان', 'حل', '##و', 'و', 'ال', '##ك', '##ل', 'ج' 'ال', '##ه', 'ش', '##وي', '##ه##', 'ب', '##س', 'ال', '##عار', 'ع']
ahmedabdelali/bert-base-qarib	['الاسعار', 'عاليه', 'شويه', 'المكان', 'حلو', 'و', 'الاكل', 'جميل', 'و', 'نظيف', 'بس']
aubmindlab/bert-base-arabertv02-twitter	['الاسعار', 'عاليه', 'شوي', '##ه', 'المكان', 'حلو', 'و', 'الاكل', 'جميل', 'و', 'نظيف', 'بس']
asafaya/bert-base-arabic	['الاسعار', 'عاليه', 'شوي', '##ه', 'المكان', 'حلو', 'و', 'الاكل', 'جميل', 'و', 'نظيف', 'بس']
lanwuwei/GigaBERT-v3-Arabic-and-English	['الاسعار', 'عاليه', 'شوي', '##ه', 'المكان', 'حلو', 'و', 'الاكل', 'جميل', 'و', 'نظيف', 'بس']
aubmindlab/araelectra-base-discriminator	['الاسعار', 'عاليه', 'شوي', '##ه', 'المكان', 'حلو', 'و', 'الاكل', 'جميل', 'و', 'نظيف', 'بس']

Fig. 7 shows the output of various Tokenizers

3.4 Classification Algorithms

Transformers first burst into the scene in that famous paper [attention is all you need]Vaswani et al. (2023) and pretty much flipped the NLP world on its head. Models built on this idea now solve tasks like text classification and sentiment analysis in surprising ways, generally nailing state-of-the-art results. Then, Various Arabic BERT models show some intriguing contrasts in how they're built. You'll see, in the table below, details that range from the type of training corpus to the size of the vocabulary and even the tokenization approach used, all of which, in most cases, play a role in how the model performs in NLP tasks. It might seem a bit all over the place, but this collection of attributes really matters, since differences like these can sometimes change a model's effectiveness in different settings. Delving into the array of BERT choices tailored for distinct Arabic NLP demands holds considerable weight in propelling computational linguistics forward. This investigation doesn't just inspire deeper dives into model behavior when corpus types and tokenization strategies are combined, but it also sets a basic stage for future Arabic NLP advancements, mirroring persistent endeavors to boost machine learning applications within this domain Fouadi et al. (2024). The study provides a complete examination of different Arabic and multilingual pretrained models' configurations. Understanding the capabilities of different pretrained language models relies on several key statistics, as explored below.

We can compare these pretrained language models by looking at things like the number of parameters they have, the type of tokenizer they use, the language(s) they're trained on, the number of layers, and their embedding dimensions. These aspects, of course, are all essential when trying to figure out how well a pretrained language model performs and its overall architecture. Now, each of these specifications is important for figuring out how effective the models will be for specific tasks and, of course, datasets. Generally speaking, a number of studies, such as Fields et al. (2024) ,Ghallab et al. (2020),Chowdhury et al. (2020) and Alammary (2022), have identified several models that tend to be particularly effective for Arabic text classification. Specifically, these include XLMRoBERTa, AraBERT, CAMELBERT, MARBERT, QARIB, ArabicBERT, DISTILBERT, GigaBERT, and ARAELECTRA. These algorithms have demonstrably outperformed more classical and deep learning approaches in various linguistic tasks. This success can largely be attributed to their ability to capture the often-subtle contextual nuances inherent in language, while also effectively handling the unique complexities presented by Arabic script below in Table 1 comparison between these pretrained language models.

Table 1 Comparison of Arabic Pre-trained Language Models

Model	**Variants**	**#Layers**	**#Heads**	**Hidden Size**	**Max Seq. Len.**	**Params**	**Tokenization**
AraBERTv0.2	MSA	12	12	768	512	110M	SentencePiece
ArabicBERT Base	MSA	12	12	768	512	110M	WordPiece
MARBERT	MSA /DA	12	12	768	512	163M	WordPiece
XLM-R Base	MSA	12	12	768	512	270M	SentencePiece
CAMeLBERT	MSA /DA/CA	–	–	–	512	–	WordPiece
QARiB	MSA /DA	12	12	768	512	110M	Seg.-Agnostic
AraBERT Twitter	MSA	12	12	768	512	110M	SentencePiece
DistilBERT Arabic	MSA	6	6	768	512	66M	–
GigaBERT V3	MSA	12	12	768	512	125M	WordPiece
AraELECTRA	MSA	12	12	768	512	136M	WordPiece

Below we will introduce these models:

3.4.1 QARIB

The Qatar Computing Research Institute (QCRI) developed this Arabic model, and it's designed to support a range of Arabic dialects and, importantly, Arabic text from social media, which poses specific linguistic challenges. To ensure a strong base, it's trained on datasets like the Arabic Gigaword Fourth Edition El-khair (2016), Open Subtitles Lison and Tiedemann (2016), plus informal text gathered from Twitter. It's been rigorously assessed on tasks like Named Entity Recognition, Sentiment Analysis, and, notably, Arabic Dialect Identification. These tests show how effective and versatile it is in handling Arabic language nuances. As some recent research indicates, mixing formal with informal text during training can be quite helpful, leading to cutting-edge results in areas like emotion detection and text classification important for today's applications Maruf et al. (2024).

3.4.2 ArabicBERT

Originating from the KUIS AI lab at Koc¸ University, Istanbul, this project presents a collection of four unique pre-trained transformer models, each meticulously crafted for Arabic. Significantly, it stands as the initial publicly accessible BERT-based pre-trained model explicitly built for Arabic, addressing a notable deficiency in available resources. The approach to training leverages masked language modeling, further refined by integrating whole word masking techniques Devlin et al. (2019). Training involved a varied blend of OSCAR data Suárez et al. (2020) and a considerable 8.2-billion-word corpus from a current Wikipedia data dump Al-Twairesh (2021). Consequently, the dataset encompasses a diverse array of Modern Standard Arabic (MSA), various dialectal forms of Arabic (DA), plus some non-Arabic terms intentionally left unfiltered, which can generally improve performance on tasks like Named Entity Recognition (NER). For flexibility and computational scaling, four size variants are available: Large, Base, Medium, and Mini.

3.4.3 DISTILBERT

DistilBERT Raiaan et al. (2024), a leaner takes on the original BERT, is quite effective for NLP. Being quicker, smaller, and more economical than BERT is what defines it. These traits make it well-suited for on-device work, particularly as real-time language capabilities become more vital. The basic architectural plan mirrors BERT, but with half the layers, and, importantly, ditching both the pooler and token-type embeddings for a more streamlined approach. The remaining layers are tuned to work well in current setups, so it does surprisingly well for its size. For developers looking to balance power and resources, it's a pretty good choice. It's trained on the same big datasets as BERT, English Wikipedia and the Toronto Book Corpus Yao and Huang (2018).

3.4.4 GigaBERT

Specifically engineered for English-to-Arabic cross-lingual transfer and adept at handling different Arabic NLP challenges, the model represents a bespoke iteration of

BERT. Its training is thorough, leveraging an extensive collection of newswire articles from the Gigaword corpus Parker et al. (2011), alongside data obtained from Wikipedia and web crawls. The training process also benefits from the integration of select Oscar dataset versions Suárez et al. (2020). From an architectural point of view, it mirrors BERT's established structure: twelve attention layers, each featuring twelve attention heads, coupled with 768 hidden dimensions, adding up to 110 million parameters in all. The model's capabilities have been carefully assessed via Arabic Information Extraction (IE) tasks, which include Part-of-Speech (POS) tagging, Named Entity Recognition (NER), Relation Extraction (RE), and Arabic Relation Learning (ARL). It has shown remarkable improvements compared to other multilingual models, like XLM-RoBERTa.

3.4.5 ARAELECTRA

Developed at the American University of Beirut by Wissam Antoun, Fady Baly, and Hazem Hajj Antoun et al. (2021), the model stands as a language model, pre-trained and focusing on Arabic. The model training leveraged a sizable Arabic corpus, including the OSCAR corpus Suárez et al. (2020), the 1.5B Words Arabic Corpus El-khair (2016), a full Arabic Wikipedia dump taken in September 2020, the OSIAN corpus Zeroual et al. (2019), and news articles from As-Safir newspaper. Speaking architecturally, the model is, a bidirectional transformer encoder. It has a base configuration with 12 layers, as well as 12 attention heads, a hidden size of 768, and an input sequence length of 512, adding up to 136 million parameters in total. It is worth noting that it achieves top results in sentiment analysis (SA), question answering (QA), and named-entity recognition (NER) for Arabic, remaining relatively compact when compared to similar models.

3.4.6 AraBERT

AraBERT is a language model, pre-trained and focused on Arabic. It's built using BERT architecture. It is trained on a varied collection of Modern Standard Arabic (MSA) texts from different sources, AraBERT exists in various versions, each designed for particular uses. Its architecture includes 768 hidden units, twelve attention heads, and a 512-token maximum sequence length, totaling 110 million parameters, a design capable of understanding complicated language structures and achieving high performance in Arabic text tasks. Neural topic modeling approaches, like those in AraBERTopic, also highlight its adaptability in identifying themes from Arabic news, strengthening its usefulness and reliability in current linguistic research HABBAT et al. (2023).

3.4.7 XLMRoBERTa

Facebook's XLMRoBERTa is notable for its impressive language support, including Arabic. Trained on Wikipedia and Common Crawl, it's quite adaptable. The Arabic data is in Modern Standard Arabic (MSA), a good base for Arabic NLP. The model has twelve layers of transformer blocks, 768 hidden units, twelve self-attention heads, and roughly 270 million parameters, which shows how complex it is.

3.4.8 CAMELBERT

CAMELBERT is a collection of BERT models with sizes and text adaptations specific to Arabic. Some CAMELBERT versions are trained on Modern Standard Arabic (MSA), dialectal Arabic (DA), and classical Arabic (CA); one is even pre-trained on a mixture of these, meeting various linguistic needs while keeping BERT-base architecture.

3.4.9 MARBERT

MARBERT is specifically for Arabic. Unlike AraBERT, it's trained on MSA and several Arabic dialects. It uses a large Twitter dataset, informing its understanding of current language. MARBERT's data volume exceeds AraBERT, generally improving performance in tasks with various Arabic dialects. MARBERT has the same architecture as Multilingual BERT, but without next-sentence prediction (NSP), further emphasizing its focus on understanding Arabic nuances Chang et al. (2024). The wide effects of models such as these are essential, influencing not just NLP but also a deeper understanding of AI across many fields Li (2024).

4 Experiments, Evaluation and Results

In this section, we'll explore the experimental setup's details, look closely at what happened with the results, and carefully check how well different pretraining models did. The methods we used in the experiments aren't just about figuring out what the pretrained language models can do. They also make sure the evaluations can be measured and compared fairly. The results we're about to discuss don't just show how these pretraining models did in a straightforward way; they also show how useful they can be in different situations and with different sets of data. Understanding this is key for seeing how they might work in the real world. By really digging into these outcomes, we can start to see the good and bad sides of the models we tested. This could help us make them even better down the road.

4.1 Hardware environment

The Google Colab platform was chosen for all code execution in this study. The extensive computing power it offers is generally necessary for both training and fine-tuning pre-trained models. Colab's high-performance GPUs, and its seamless data integration capabilities with Google Drive, guided this selection. The storage and retrieval of datasets, trained models, and output results are streamlined through this integration. Researchers can also employ tools such as TensorFlow, HuggingFace and Keras, which are essential for effective model implementation. Using such powerful platforms is vital for current NLP tasks, as noted in Ram (2023). The specifications used during the code execution are presented here. GPU: NVIDIA-SMI Tesla T4 (15.0 GB) RAM: 12.7 GB STORGE: 112.6 GB

4.2 Software environment

The software environment is made up of Python 3.8, which is installed in Google Colab, in accordance with the findings highlighted in Alqahtani and Alothaim (2022), Vajjala et al. (2020) concerning effective tools for managing text data. Further than core libraries, several other tools and libraries were used to ensure robust performance and comprehensive functionality.

Below are some of the Additional Tools used and libraries: • TensorFlow and Keras: used for finetuning pretrained models • HuggingFace Transformers: used to load pre-trained models from HuggingFace • Nltk and re: used for preprocessing and cleaning textual data • NumPy and Pandas: used to handle numerical computations like matrices • Matplotlib and wordcloud: used for visualization and analyzing • Bidi, Arabic reshaper: used for handling Arabic Text The dataset was mounted from google drive to google Colab as Excel file and split to 60% training ,20% for validation and testing, respectively.

4.3 Hyperparameters configuration

Our study delves into fine-tuning pretrained language models, using both grid search and random search. This helps us pinpoint those crucial hyperparameter values needed for fine-tuning, to get the best performance and a good overall evaluation. Now, about those hyperparameters: We're using the Adam optimizer, with a learning rate set to (1e-6). As Jayakumar et al. (2025) notes, this is a solid choice for helping things converge nicely during training. Then, we've got a dropout rate of (0.2). Dropout, as you probably know, helps keep our deep learning models from overfitting by switching off some neurons randomly during training . Sparse Categorical Cross-Entropy, That's our loss function. A common one for when we're tackling multi-class classification. The model sees data in batches of 128 at a time. Batch size matters, and this setting improves training efficiency and convergence. Plus, early stopping is in place to block overfitting. It watches the validation loss, with (monitor=val_loss, min_delta=0, a patience of 5, verbose=1, and mode=auto). Training can go on for 50 epochs, but models will stop learning after a set number of epochs, which you can see in table 2. So, with these carefully chosen hyperparameters, we're looking to make our trained models more robust and perform better, adding to the progress in model training methods.

4.4 Performance metrics

Pretrained language models get checked out in lots of different ways when it comes to handling classification tasks, and in recent work you'll see metrics like accuracy, precision, recall, the F1-score, and even the Receiver Operating Characteristic (ROC) curve thrown around quite a bit Selim et al. (2024) .Accuracy is the simple idea of taking the number of correct predictions and dividing it by the total predictions made but be warned, when the dataset is imbalanced, this number can mislead, since some classes dominate over others. Several studies Attieh and Hassan (2022),Vajjala et al. (2020). have noted that you really need to look at the context when reading accuracy scores. Precision comes next. It's just the fraction of positive predictions that are

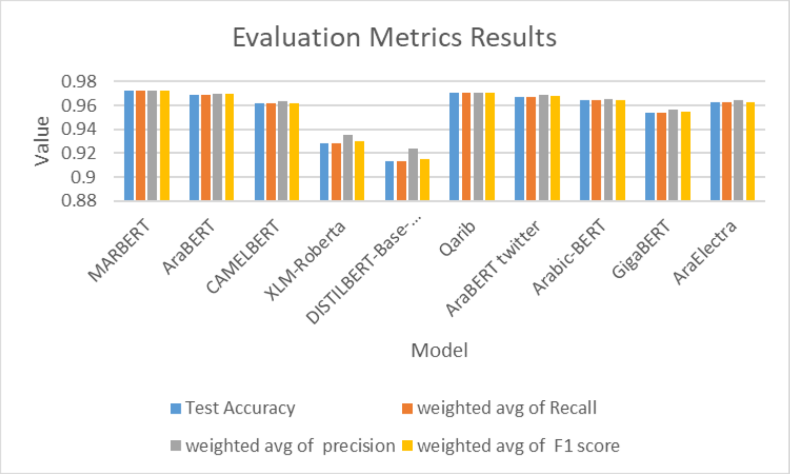


Fig. 8 shows the Evaluation Metrics Results of Finetuned models

:

actually on target, computed as true positives divided by the sum of true and false positives. Then recall jumps in, measuring how many of the actual positives get caught by the model by comparing true positives to the total actual positives. These two ideas keep repeating because they each tell you a bit of the story. When you blend them together, you get the F1-score, a weighted average that gives you one number to understand the balance between precision and recall. A higher F1-score means the model performs better. Then there's the ROC curve, a visual tool that lays out the trade-off between the true positive rate (TPR) and false positive rate (FPR) across different threshold settings. Typically, a rising area under this curve means the model is doing a better job at flagging true cases correctly. This graphical approach also sheds light on the model's ability to tell classes apart, adding another layer to the overall performance check. For a more systematic view, we rely on these mathematical forms: Accuracy = (TP + TN) / (TP + TN + FP + FN), Precision = TP / (TP + FP), Recall = TP / (TP + FN), and F1-score = 2 × (Precision × Recall) / (Precision + Recall). Here, TP stands for those true positives instances correctly flagged as the target and TN covers those true negatives, or the cases rightly left out.

4.5 Evaluation and discussion

In this section, we delve into a comprehensive comparison of the results from finetuning ten distinct pretrained language models, as shown in Figure 8 and also Table 2. We will analyze diverse factors that significantly influence the performance of these experiments. This analysis is crucial for achieving elevated performance levels specifically in classifying Arabic text, a progressively more important area of study. (Fig 6) outlines the Evaluation Metrics Results of these finetuned pretrained language models, highlighting the key performance indicators reflecting each approach's effectiveness.

- The MARBERT model demonstrates robust performance, surpassing other models in accuracy and most testing metrics, suggesting remarkable generalization. Its

Table 2 Performance comparison of Finetuning Arabic pretrained Language models

Model	**Model name on Hugging-Face**	**Number of Epochs**	**Avg. Training Time per Epoch (min:sec)**	**Size (MB)**	**Loss %**	**Accuracy %**	**Recall %**	**Precision %**	**F1 score %**	**Testing Time / Sam-ple (ms)**
MARBERT	UBC-NLP/ MARBERT	17	03:56	621.20	**0.099**	**97.23**	**97.23**	**97.25**	**97.24**	3.061
AraBERT	aubmindlab /bert-base-arabertv02	34	03:42	515.73	0.103	96.92	96.92	97.01	96.94	4.223
CAMELBERT	CAMeL-Lab /bert-base-arabic-camelbert-da-sentiment	22	03:46	**416.12**	0.138	96.19	96.19	96.31	96.21	4.292
XLM-Roberta	FacebookAI /xlm-roberta-base	**15**	03:59	1060.66	0.206	92.86	92.86	93.56	92.97	2.881
DISTILBERT	lxyuan /distilbert-base-multilingual-cased-sentiments-student	34	**02:04**	516.23	0.255	91.33	91.33	92.39	91.50	**2.114**
Qarib	ahmedabdelali /bert-base-qarib	22	03:52	515.73	0.111	97.04	97.04	97.09	97.05	4.231
AraBERT-twitter	aubmindlab /bert-base-arabertv02-twitter	20	03:45	515.73	0.107	96.73	96.73	96.85	96.75	2.838
Arabic-BERT	asafaya /bert-base-arabic	29	03:43	421.98	0.128	96.43	96.43	96.50	96.45	2.879
GigaBERT	lanwuwei /GigaBERT-v3-Arabic-and-English	22	03:49	474.71	0.140	95.39	95.39	95.64	95.43	4.241
AraELECTRA	aubmindlab /araelectra-base-discriminator	17	04:06	515.73	0.124	96.24	96.24	96.40	96.27	2.973

accuracy peaks at 97.23%, balancing precision 97.24% and recall 97.22% for an F1 score of 97.27%. Learning concluded after 17 epochs, consuming 4012.81 seconds and 621.19 MB of memory; average review prediction time was 3.0 milliseconds. • Similarly, QARIB achieves notable results, attaining 97.03% accuracy, with precision at 97.09% and recall at 97.04%, resulting in an F1 score of 97.05%. It hit these marks after 22 epochs, using 5105.41 seconds, then stopped learning; storage required 515.72 MB, and average review prediction time was 4.2 milliseconds. • AraBERT Base v2 maintains high accuracy, specifically 96.92%, accompanied by a precision of 97.01%, recall of 96.92%, and F1 score of 96.94%. These values were achieved in 34 epochs, using 7552.06 seconds, then halting, utilizing 515.73 MB for storage, prediction time averaged 4.2 milliseconds. • The AraBERT model fine-tuned on tweets is also a strong performer, showing an accuracy of 96.73%, precision of 96.5%, and recall of 96.73%, while its F1 score lands at 96.75%. Training took 20 epochs, consuming 4491.51 seconds, after which it stopped learning, requiring 515.73 MB for storage; review prediction time was about 2.8 milliseconds. • Arabic-BERT achieves an accuracy around 96.43%, with precision held at 96.5% and recall at 96.42%, yielding a 96.45% F1 score. This was accomplished in 29 epochs using 6471.3587 seconds, eventually stopping with a memory footprint of 421.98 MB, while each review took 2.8 milliseconds to predict. • AraELECTRA shows about 96.24% accuracy, precision of 96.27%, and recall held at 96.24%, to get an F1 score of 96.27%. Satisfactory, these results came after 17 epochs, 4176.512 seconds, stopping and taking 515.73 MB of storage, averaging about 3 milliseconds to predict each review. • CAMELBERT showcases solid accuracy at approximately 96.19%, precision reported as 96.31% and recall also 96.19%, sustaining an F1 score of 96.21%. After 22 epochs (4971.005 seconds), learning stopped, and this model used 416.12 MB of storage, predicting reviews in roughly 4.3 milliseconds. • GigaBERT, too, achieves a commendable accuracy around 95.39%; it had precision of 95.64% and recall of 95.39%, so its F1 score was 95.43%. These followed 22 epochs requiring 5033.1134 seconds, after which the model ceased learning and utilized 474.71 MB for storage, with the average review being predicted in 4.2 milliseconds. • XLM-Roberta reached an accuracy around 92.86%, with precision being about 93.56% and recall being approximately 92.86%, and an F1 score of 92.97%. Those results, after 15 epochs and using up to 3582.10 seconds, at last it stopped, requiring some 1060.66 MB of storage, but each review took only approximately 2.9 milliseconds to predict. • Lastly DISTILBERT shows a somewhat suboptimal performance if you look at it, showing lower accuracy reported at about 91.33%, and with precision 92.39%, recall 91.33%, and an F1 score of 91.51%. Achieved in 34 epochs and utilizing 4207.76 seconds, and requiring 516.23 MB, and a prediction time of 2.1 milliseconds per single review. Table 2 shows that the fastest model in finetuning each epoch time is DISTILBERT. And fastest model in terms of predicting each review is DISTILBERT as it just has six hidden layers but for models that have twelve hidden layers (AraBERT v2) comes as fastest model in training each model and the fastest model for average for predicting one review is AraBERT v2 twitter. The benefits of using language specific pretrained language models are observed clearly as models (MARBERT) (AraBERT) (QARIB) performance surpass the performance of models which are trained on multilingual models such as (XLM-RoBERTa). The results overall highlight benefits derived from fine-tuning in classification tasks based on the Arabic text, across evaluation metrics, compared to results reported previously. emphasizing importance of embedding and transformer architecture. The effectiveness is notable, capturing patterns in language to get solid performance to detect review polarity.

5 Limitation

In this section, we'll touch on some of the hurdles encountered while fine-tuning our dataset specifically, Google Maps reviews for Egyptian restaurants. A close look at misclassified reviews revealed common error sources, notably the persistent challenges facing Arabic sentiment analysis researchers. It's worth noting that user reviews often blend Modern Standard Arabic (MSA) with various Arabic dialects (AD), and existing Natural Language Processing (NLP) tools sometimes struggle to effectively handle this linguistic diversity. Another issue, the very nature of corpora used for pre-trained language models. For example, a model trained purely on MSA may not perform optimally with dialectical content. The GigaBERT model, for instance, doesn't quite surpass an F1 score of 95.43%. Models like MARBERT, however, hit 97.27% on the same data. This is likely because GigaBERT trained only on MSA, while MARBERT incorporated both MSA and dialectal Arabic. Aligning pre-training corpora with fine-tuning task corpora is truly important Alqahtani and Alothaim (2022). Thus, a major constraint is the scarcity of large, diverse dialectal corpora for both pre-training and fine-tuning across tasks. Furthermore, the inherent ambiguity in Arabic complicates matters, as word meanings shift with context, diacritics, and even the author's underlying sentiment. Dealing with user reviews adds layers of complexity. Think emojis, which are crucial for conveying emotion in reviews across many languages, or the use of Arabizi (writing Arabic using Latin letters). The meanings of these elements are not always straightforward, varying with occasion, culture, or region. Moreover, Arabic is morphologically rich; verbs and nouns have numerous conjugations. Because reviewers aren't always language experts, reviews can have misspellings, informal language, unique symbols or abbreviations relating to particular places, cultures, or restaurants, along with potential human biases, too. Dataset labeling also presents a hurdle; some reviews are unintentionally misleading, ambiguous, or even two-sided. Let us not forget the computational resources and time needed, potentially heightening these challenges. These problems can be seen to amplify when dealing with user reviews, largely because they frequently deviate from writing conventions, if truth be told.

6 Conclusion

The investigation at hand delves into how Arabic pre-trained models can be leveraged for Arabic text classification, with a specific emphasis on dissecting user-generated reviews. Notably, we've curated a dataset consisting of 51,808 annotated Arabic reviews. These reviews, obtained from Google Maps, offer insights from restaurant patrons. It's worth pointing out that this collection stands as the most expansive compilation of Arabic restaurant reviews sourced from Google Maps. Our sentiment analysis project aims to use Transformer-based models to classify the sentiment within these reviews effectively. In doing so, existing Arabic models were finetuned using our dataset to discern between positive and negative restaurant evaluations. Recognizing that the dataset wasn't balanced, we decided that accuracy alone wouldn't suffice for judging performance. Therefore, metrics like recall, precision, F1-score, along with the ROC curve, were integrated to solidify our assessment. Careful experimentation was carried out to pinpoint hyperparameters that would yield optimal model performance. The findings suggest significant strides in Arabic dataset analysis and Arabic text comprehension. It is important to remember that selecting the right tools, libraries, preprocessing steps, and hyperparameters is crucial for improving performance metrics. We also present a comparative look at pre-trained Arabic models, shedding light on their respective advantages and disadvantages. While prior studies have thoroughly explored classical and deep learning models, as noted in earlier reviews, we've steered clear of those paths to concentrate on the specific impact of transformer-based methods. In sentiment analysis, models have shown remarkable abilities, often achieving state-of-the-art results. Specifically, MARBERT has emerged as a top performer. Its accuracy reached 97.23%, recall was 97.22%, precision hit 97.24%, and the F1-score was 97.27% – quite impressive results. This research helps those in charge better grasp and analyze public reviews and experiences. It also reduces the reliance on traditional methods, which often take a lot of work and resources. The framework presented here can be applied worldwide, allowing for the assessment of user feedback in various public and private settings, like libraries, transport, or parks. However, it's crucial to remember that choosing the best model isn't just about accuracy. Decision-makers should also consider things such as how fast it works, and the computing power needed, leading to better sentiment analysis models and increased interpretability of results overall.

7 Future work

For the future, our plans involve expanding the variety of preprocessing methods and algorithms we use. This is with an eye toward improving our models' performance, something that aligns with the growing importance placed on methodologies that work well in NLP technology. In the next steps, we plan to improve how well the pre-trained models perform in many ways. For example, this could include growing the datasets, experimenting with a broader range of datasets created by users, including more features for classification work (for example, location) and aiming to ensure outcomes offer more in-depth details about the restaurant rather than just classifying input as positive or negative. We are also interested in adopting additional pre-trained models that are coming out of current research. Our intentions involve applying Large Language Models (LLMs) to develop personalized replies to user feedback in a manner that is more engaging.

Data Availability

The Dataset used in this study and the all codes are available on : https://github.com/menaahmed22/Sentiment-Analysis-of-Arabic-Reviews-of-Restaurants-Using-Pretrained-Language-Models .

References

Abaimov S, Bianchi G (2021) A survey on the application of deep learning for code injection detection. Array 11:100077. https://doi.org/10.1016/j.array.2021.100077

Abdelali A, Hassan S, Mubarak H, et al (2021) Pre-training bert on arabic tweets: Practical considerations. URL https://arxiv.org/abs/2102.10684, arXiv:2102.10684

Abdul-Mageed M, Elmadany A, Nagoudi EMB (2021) ARBERT & MARBERT: Deep bidirectional transformers for Arabic. In: Zong C, Xia F, Li W, et al (eds) Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Association for Computational Linguistics, Online, pp 7088–7105, https://doi.org/10.18653/v1/2021.acl-long.551, URL https://aclanthology.org/2021.acl-long.551/

Abdul-Rahman G, Haleem N, Zwitter A (2025) A data science approach to mitigating data challenges in serious gaming. Discover Data 3:3. https://doi.org/10.1007/s44248-025-00023-9

Abu Farha I, Magdy W (2021) Benchmarking transformer-based language models for Arabic sentiment and sarcasm detection. In: Habash N, Bouamor H, Hajj H, et al (eds) Proceedings of the Sixth Arabic Natural Language Processing Workshop. Association for Computational Linguistics, Kyiv, Ukraine (Virtual), pp 21–31, URL https://aclanthology.org/2021.wanlp-1.3/

Addi HA, Ezzahir R (2024) Supervised classifiers and keyword extraction methods for text classification in arabic. In: 2024 IEEE 12th International Symposium on Signal, Image, Video and Communications (ISIVC). IEEE, pp 1–6, https://doi.org/10.1109/ISIVC61350.2024.10577782

Akkaya M, Özlem Özçevik, Tepe E (2024) A machine learning application to google maps reviews as a participatory planning tool. International Journal of Urban Sciences 28:379–402. https://doi.org/10.1080/12265934.2024.2320916

Al-Twairesh N (2021) The evolution of language models applied to emotion analysis of arabic tweets. Information 12:84. https://doi.org/10.3390/info12020084

Alammary AS (2022) Bert models for arabic text classification: A systematic review. Applied Sciences 12:5720. https://doi.org/10.3390/app12115720

Alkaoud M, Syed M (2020) On the importance of tokenization in Arabic embedding models. In: Zitouni I, Abdul-Mageed M, Bouamor H, et al (eds) Proceedings of the Fifth Arabic Natural Language Processing Workshop. Association for Computational Linguistics, Barcelona, Spain (Online), pp 119–129, URL https://aclanthology.org/2020.wanlp-1.11/

Almurqren L, Hodgson R, Cristea A (2024) Arabic text sentiment analysis: Reinforcing human-performed surveys with wider topic analysis. URL https://arxiv.org/abs/2403.01921, arXiv:2403.01921

Alqahtani G, Alothaim A (2022) Emotion analysis of arabic tweets: Language models and available resources. Frontiers in Artificial Intelligence 5. https://doi.org/10.3389/frai.2022.843038

Alqarni A, Rahman A (2023) Arabic tweets-based sentiment analysis to investigate the impact of covid-19 in ksa: A deep learning approach. Big Data and Cognitive Computing 7:16. https://doi.org/10.3390/bdcc7010016

Antoun W, Baly F, Hajj H (2020) AraBERT: Transformer-based model for Arabic language understanding. In: Al-Khalifa H, Magdy W, Darwish K, et al (eds) Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection. European Language Resource Association, Marseille, France, pp 9–15, URL https://aclanthology.org/2020.osact-1.2/

Antoun W, Baly F, Hajj H (2021) AraELECTRA: Pre-training text discriminators for Arabic language understanding. In: Habash N, Bouamor H, Hajj H, et al (eds) Proceedings of the Sixth Arabic Natural Language Processing Workshop. Association for Computational Linguistics, Kyiv, Ukraine (Virtual), pp 191–195, URL https://aclanthology.org/2021.wanlp-1.20/

Attieh J, Hassan F (2022) Arabic dialect identification and sentiment classification using transformer-based models. In: Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP). Association for Computational Linguistics, pp 485–490, https://doi.org/10.18653/v1/2022.wanlp-1.54

Brown PF, Della Pietra VJ, deSouza PV, et al (1992) Class-based n-gram models of natural language. Computational Linguistics 18(4):467–480. URL https://aclanthology.org/J92-4003/

Chang Y, Wang X, Wang J, et al (2024) A survey on evaluation of large language models. ACM Transactions on Intelligent Systems and Technology 15:1–45. https://doi.org/10.1145/3641289

Chen CC, Chang CC (2024) Evaluating public library services in taiwan through user-generated content: Analyzing google maps reviews. Electronics 13:2393. https://doi.org/10.3390/electronics13122393

Chouikhi H, Chniter H, Jarray F (2021) Arabic Sentiment Analysis Using BERT Model, Springer International Publishing, Cham, pp 621–632

Chowdhury SA, Abdelali A, Darwish K, et al (2020) Improving Arabic text categorization using transformer training diversification. In: Zitouni I, Abdul-Mageed M, Bouamor H, et al (eds) Proceedings of the Fifth Arabic Natural Language Processing Workshop. Association for Computational Linguistics, Barcelona, Spain (Online), pp 226–236, URL https://aclanthology.org/2020.wanlp-1.21/

Conneau A, Khandelwal K, Goyal N, et al (2020) Unsupervised cross-lingual representation learning at scale. In: Jurafsky D, Chai J, Schluter N, et al (eds) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Online, pp 8440–8451, https://doi.org/10.18653/v1/2020.acl-main.747, URL https://aclanthology.org/2020.acl-main.747/

Das M, K. S, Alphonse PJA (2023) A comparative study on tf-idf feature weighting method and its analysis using unstructured dataset. URL https://arxiv.org/abs/2308.04037, arXiv:2308.04037

Devlin J, Chang MW, Lee K, et al (2019) Bert: Pre-training of deep bidirectional transformers for language understanding. URL https://arxiv.org/abs/1810.04805, arXiv:1810.04805

Dwivedi YK, Ismagilova E, Hughes DL, et al (2021) Setting the future of digital and social media marketing research: Perspectives and research propositions. International Journal of Information Management 59:102168. https://doi.org/10.1016/j.ijinfomgt.2020.102168

Dwivedi YK, Kshetri N, Hughes L, et al (2023) Opinion paper: "so what if chatgpt wrote it?" multidisciplinary perspectives on opportunities, challenges and implications of generative conversational ai for research, practice and policy. International Journal of Information Management 71:102642. https://doi.org/10.1016/j.ijinfomgt.2023.102642

El-khair IA (2016) 1.5 billion words arabic corpus. URL https://arxiv.org/abs/1611.04033, arXiv:1611.04033

Fields J, Chovanec K, Madiraju P (2024) A survey of text classification with transformers: How wide? how large? how long? how accurate? how expensive? how safe? IEEE Access 12:6518–6531. https://doi.org/10.1109/ACCESS.2024.3349952

Fouadi H, Moubtahij HE, Lamtougui H, et al (2024) Bert-based models for classifying multi-dialect arabic texts. IAES International Journal of Artificial Intelligence (IJ-AI) 13:3437. https://doi.org/10.11591/ijai.v13.i3.pp3437-3446

Galal MA, Yousef AH, Zayed HH, et al (2024) Arabic sarcasm detection: An enhanced fine-tuned language model approach. Ain Shams Engineering Journal 15:102736. https://doi.org/10.1016/j.asej.2024.102736

Ghallab A, Mohsen A, Ali Y (2020) Arabic sentiment analysis: A systematic literature review. Applied Computational Intelligence and Soft Computing 2020:1–21. https://doi.org/10.1155/2020/7403128

Guan X, Treude C (2024) Enhancing source code representations for deep learning with static analysis. URL https://arxiv.org/abs/2402.09557, arXiv:2402.09557

HABBAT N, ANOUN H, HASSOUNI L (2023) Arabertopic: A neural topic modeling approach for news extraction from arabic facebook pages using pre-trained bert transformer model. International Journal of Computing and Digital Systems 14:1–8. https://doi.org/10.12785/ijcds/140101

Hakami SAA, Hendley R, Smith P (2022) A context-free Arabic emoji sentiment lexicon (CF-Arab-ESL). In: Al-Khalifa H, Elsayed T, Mubarak H, et al (eds) Proceedinsg of the 5th Workshop on Open-Source Arabic Corpora and Processing Tools with Shared Tasks on Qur'an QA and Fine-Grained Hate Speech Detection. European Language Resources Association, Marseille, France, pp 51–59, URL https://aclanthology.org/2022.osact-1.6/

Inoue G, Alhafni B, Baimukan N, et al (2021) The interplay of variant, size, and task type in Arabic pre-trained language models. In: Habash N, Bouamor H, Hajj H, et al (eds) Proceedings of the Sixth Arabic Natural Language Processing Workshop. Association for Computational Linguistics, Kyiv, Ukraine (Virtual), pp 92–104, URL https://aclanthology.org/2021.wanlp-1.10/

Jayakumar VM, Rajakumari R, Alapati PR, et al (2025) Enhancing english language assessment in educational settings using natural language processing techniques. In: 2025 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC). IEEE, pp 438–443, https://doi.org/10.1109/ISACC65211.2025.10969428

Kasongo SM, Sun Y (2020) Performance analysis of intrusion detection systems using a feature selection method on the unsw-nb15 dataset. Journal of Big Data 7:105. https://doi.org/10.1186/s40537-020-00379-6

Koshiry AME, Eliwa EHI, El-Hafeez TA, et al (2023) Arabic toxic tweet classification: Leveraging the arabert model. Big Data and Cognitive Computing 7:170. https://doi.org/10.3390/bdcc7040170

Lan W, Chen Y, Xu W, et al (2020) An empirical study of pre-trained transformers for arabic information extraction. URL https://arxiv.org/abs/2004.14519, arXiv:2004.14519

Li J (ed) (2024) Advances in Sentiment Analysis - Techniques, Applications, and Challenges, vol 22. IntechOpen, https://doi.org/10.5772/intechopen.111293

Libovický J, Rosa R, Fraser A (2019) How language-neutral is multilingual bert? URL https://arxiv.org/abs/1911.03310, arXiv:1911.03310

Lison P, Tiedemann J (2016) OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In: Calzolari N, Choukri K, Declerck T, et al (eds)

Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). European Language Resources Association (ELRA), Portorož, Slovenia, pp 923–929, URL https://aclanthology.org/L16-1147/

Maruf AA, Khanam F, Haque MM, et al (2024) Challenges and opportunities of text-based emotion detection: A survey. IEEE Access 12:18416–18450. https://doi.org/10.1109/ACCESS.2024.3356357

Mikolov T, Chen K, Corrado G, et al (2013) Efficient estimation of word representations in vector space. URL https://arxiv.org/abs/1301.3781, arXiv:1301.3781

Minaee S, Kalchbrenner N, Cambria E, et al (2021) Deep learning based text classification: A comprehensive review. URL https://arxiv.org/abs/2004.03705, arXiv:2004.03705

Mousa A, Shahin I, Nassif AB, et al (2024) Detection of arabic offensive language in social media using machine learning models. Intelligent Systems with Applications 22:200376. https://doi.org/10.1016/j.iswa.2024.200376

Musleh DA, Alkhwaja I, Alkhwaja A, et al (2023) Arabic sentiment analysis of youtube comments: Nlp-based machine learning approaches for content evaluation. Big Data and Cognitive Computing 7:127. https://doi.org/10.3390/bdcc7030127

Omar A, Mahmoud TM, Abd-El-Hafeez T, et al (2021) Multi-label arabic text classification in online social networks. Information Systems 100:101785. https://doi.org/10.1016/j.is.2021.101785

Ouassil MA, Jebbari M, Rachidi R, et al (2024) Enhancing arabic text readability assessment: A combined bert and bilstm approach. In: 2024 International Conference on Circuit, Systems and Communication (ICCSC). IEEE, pp 1–7, https://doi.org/10.1109/ICCSC62074.2024.10616953

Oueslati O, Cambria E, HajHmida MB, et al (2020) A review of sentiment analysis research in arabic language. Future Generation Computer Systems 112:408–430. https://doi.org/10.1016/j.future.2020.05.034

Parker R, Graff D, Kong J, et al (2011) English gigaword fifth edition

Pennington J, Socher R, Manning C (2014) Glove: Global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, pp 1532–1543, https://doi.org/10.3115/v1/D14-1162

Qader WA, Ameen MM, Ahmed BI (2019) An overview of bag of words;importance, implementation, applications, and challenges. In: 2019 International Engineering Conference (IEC). IEEE, pp 200–204, https://doi.org/10.1109/IEC47844.2019.8950616

Raiaan MAK, Mukta MSH, Fatema K, et al (2024) A review on large language models: Architectures, applications, taxonomies, open issues and challenges. IEEE Access 12:26839–26874. https://doi.org/10.1109/ACCESS.2024.3365742

Ram (2023) Advancing cloud data analytics and chatbots through machine learning technology: Key recommendations. International Scientific Journal for Research 5(5). URL https://isjr.co.in/index.php/ISJR/article/view/141

Safaya A, Abdullatif M, Yuret D (2020) KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media. In: Herbelot A, Zhu X, Palmer A, et al (eds) Proceedings of the Fourteenth Workshop on Semantic Evaluation. International Committee for Computational Linguistics, Barcelona (online), pp 2054–2059, https://doi.org/10.18653/v1/2020.semeval-1.271, URL https://aclanthology.org/2020.semeval-1.271/

Sanh V, Debut L, Chaumond J, et al (2020) Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. URL https://arxiv.org/abs/1910.01108, arXiv:1910.01108

Selim A, Ali I, Ristevski B (2024) University information system's impact on academic performance: A comprehensive logistic regression analysis with principal component analysis and performance metrics. TEM Journal pp 1589–1598. https://doi.org/10.18421/TEM132-72

Shin B, Ryu S, Kim Y, et al (2022) Analysis on review data of restaurants in google maps through text mining: Focusing on sentiment analysis. Journal of Multimedia Information System 9:61–68. https://doi.org/10.33851/JMIS.2022.9.1.61

Suárez PJO, Romary L, Sagot B (2020) A monolingual approach to contextualized word embeddings for mid-resource languages. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp 1703–1714, https://doi.org/10.18653/v1/2020.acl-main.156

Vajjala S, Majumder B, Gupta A, et al (2020) Practical Natural Language Processing. O'Reilly

Vaswani A, Shazeer N, Parmar N, et al (2023) Attention is all you need. URL https://arxiv.org/abs/1706.03762, arXiv:1706.03762

Wang G, Lu J, Choi KS, et al (2020) A transfer-based additive ls-svm classifier for handling missing data. IEEE Transactions on Cybernetics 50:739–752. https://doi.org/10.1109/TCYB.2018.2872800

Yao W, Huang R (2018) Temporal event knowledge acquisition via identifying narratives. arXiv preprint arXiv:180510956

Zeroual I, Goldhahn D, Eckart T, et al (2019) Osian: Open source international arabic news corpus - preparation and integration into the clarin-infrastructure. In: Proceedings of the Fourth Arabic Natural Language Processing Workshop. Association for Computational Linguistics, pp 175–182, https://doi.org/10.18653/v1/W19-4619