



A Comparative Evaluation of Speech-Text Effective Approaches for Arabic Sentiment Analysis

تقييم مقارن للمقاربات الفعالة لتحليل المشاعر العربية

by

MARIAM SUHAIL HADEED

**Dissertation submitted in partial fulfilment
of the requirements for the degree of
MSc INFORMATICS**

at

The British University in Dubai

June 2024

Declaration

I warrant that the content of this research is the direct result of my own work and that any use made in it of published or unpublished copyright material falls within the limits permitted by international copyright conventions.

I understand that a copy of my research will be deposited in the University Library for permanent retention.

I hereby agree that the contents of this dissertation for which I am author and copyright holder may be copied and distributed by The British University in Dubai for the purposes of research, private study or education and that The British University in Dubai may recover from purchasers the costs incurred in such copying and distribution, where appropriate.

I understand that The British University in Dubai may make a digital copy available in the institutional repository.

I understand that I may apply to the University to retain the right to withhold or to restrict access to my thesis for a period which shall not normally exceed four calendar years from the congregation at which the degree is conferred, the length of the period to be specified in the application, together with the precise reasons for making that application.



Signature of the student

Copyright and Information to Users

The author whose copyright is declared on the title page of the work has granted to the British University in Dubai the right to lend his/her research work to users of its library and as an open source and to make partial or single copies for educational and research use.

The author has also granted permission to the University to keep or make a digital copy for similar use and for the purpose of preservation of the work digitally.

Multiple copying of this work for scholarly purposes may be granted by either the author, the Registrar or the Dean only.

Copying for financial gain shall only be allowed with the author's express permission.

Any use of this work in whole or in part shall respect the moral rights of the author to be acknowledged and to reflect in good faith and without detriment the meaning of the content, and the original authorship.

Abstract

In recent years, the pervasive use of social media platforms has elevated the significance of users' emotions, ranging from satisfaction to anxiety, within digital discourse. This trend is particularly pronounced in the Arabic-speaking digital landscape, where the language holds a prominent presence across various communication platforms and social media networks. Effectively interpreting sentiment voiced in Arabic speech and script poses a significant difficulty owing to several elements. These consist of the ingrained sophistication of Arabic morphology, the scarcity of comprehensive Arabic corpora, and the divergence of Arabic accents and dialects. Conventional methods for sentiment analysis have struggled to provide accurate results in this context, prompting the exploration of deep learning techniques as a potential solution. This study tackles the challenge of Arabic sentiment analysis, encompassing modern standard Arabic as well as dialectal Arabic, by examining the effectiveness of assorted deep learning models in discerning sentiments expressed in both speech and its corresponding transcript data. Specifically, the study compares Arabic speech sentiment analysis with its transcript analysis, employing CNN, LSTM, BI-LSTM, and GRU models. A diverse dataset comprising Arabic speech and text samples, encompassing positive, negative, and neutral sentiments, was meticulously curated for this purpose. Each deep learning model was trained and evaluated separately on both speech and text data to assess its ability to discern sentiment nuances across these modalities. The results yielded insightful findings regarding the relative performance of CNN, LSTM, BI-LSTM, and GRU models across speech and text datasets. For Arabic speech sentiment analysis, BI-LSTM emerged as the most effective model, achieving an impressive accuracy rate of 89%, followed closely by CNN at 84%. In contrast, for Arabic text sentiment analysis, GRU and CNN techniques outperformed LSTM and BI-LSTM, achieving accuracy rates of 73% and 72%, respectively. Overall, this study contributes valuable insights into the domain of Arabic

sentiment analysis, shedding light on the comparative effectiveness of CNN, LSTM, BI-LSTM, and GRU models in analysing sentiments expressed through both speech and text data. These findings hold significant implications for scholars and experts seeking to develop efficient emotion evaluation systems tailored to Arabic language contexts.

Keywords: Arabic speech analysis, Arabic text analysis, sentiment analysis, Deep learning.

ملخص

في السنوات الأخيرة، أدى الانتشار الواسع لمنصات التواصل الاجتماعي إلى إبراز أهمية مشاعر المستخدمين، بدءًا من مشاعر الرضا وصولًا إلى القلق، في تشكيل الخطاب الرقمي. ويتجلى هذا بشكل خاص في البيئة الرقمية الناطقة باللغة العربية، حيث تتمتع اللغة بحضور قوي على مختلف المنصات ووسائل الاتصال. غير أن تحليل المشاعر في اللغة العربية، سواء في النصوص أو الكلام، يمثل تحديًا كبيرًا نتيجة لعدة عوامل، منها البنية الصرفية المعقدة، وندرة المدونات العربية الشاملة، وتعدد اللهجات الإقليمية. وقد عجزت الأساليب التقليدية لتحليل المشاعر عن تحقيق نتائج دقيقة في هذا السياق، مما أدى إلى الاعتماد على تقنيات التعلم العميق كبديل واعد. تهدف هذه الدراسة إلى معالجة التحديات المرتبطة بتحليل المشاعر باللغة العربية، سواء في الفصحى أو اللهجات العامية، من خلال تقييم فعالية نماذج متعددة من التعلم العميق في التمييز بين المشاعر المعبر عنها في الكلام والنصوص المقابلة لها. وتحديدًا، تقارن الدراسة بين أداء نماذج CNN و LSTM و Bi-LSTM و GRU في تحليل مشاعر الكلام والنصوص العربية. ولتحقيق ذلك، تم إعداد مجموعة بيانات متنوعة بعناية، تضم عينات من الكلام والنصوص العربية تتضمن مشاعر إيجابية وسلبية ومحايدة. جرى تدريب وتقييم كل نموذج على حدة لكل من بيانات الكلام والنص، لقياس قدرته على التقاط الفروق الدقيقة في المشاعر. أظهرت النتائج فروقًا واضحة في الأداء بين النماذج. ففي تحليل مشاعر الكلام العربي، تفوق نموذج Bi-LSTM محققًا دقة بلغت 89%، تلاه نموذج CNN بنسبة 84%. أما في تحليل النصوص، فقد سجل نموذجا GRU و CNN أداءً أعلى من LSTM و Bi-LSTM، بنسبة دقة وصلت إلى 73% و 72% على التوالي. تُقدم هذه الدراسة رؤى مهمة في مجال تحليل المشاعر باللغة العربية، موضحةً نقاط القوة النسبية لنماذج CNN و LSTM و Bi-LSTM و GRU في التعامل مع بيانات الكلام والنص. وتساهم هذه النتائج في دعم تطوير أنظمة فعالة لتحليل المشاعر، مصممة خصيصًا لتناسب خصوصية اللغة العربية وسياقاتها المختلفة.

الكلمات الأساسية: تحليل الكلام العربي، تحليل النص العربي، تحليل المشاعر، التعلم العميق.

Dedication

To my parents, conveying the depth of my gratitude is a challenge. Their words echo within me, resonating with my thoughts and emotions. It is their constant support and prompt that support my insistence to pursue doctoral studies. Throughout my master's studies, my parents, alongside my sisters and brothers, have been a continuous resource of support, prompting, and empathy. I am profoundly thankful for their love, care, and uplifting words, which have enabled me to defeat constraints and reach my aspirations.

Acknowledgements

I extend my gratitude to the entire community at the British University in Dubai (BUiD) for granting me the opportunity to delve into the field of informatics, which has enriched my knowledge immensely.

I am also grateful to Dr. Manar Alkhatib for overseeing this dissertation and facilitating its completion. Additionally, I extend my thanks to Dr Khaled Shaalan for his guidance and support throughout this research endeavor. Their assistance has been instrumental in producing a research paper of this caliber.

Finally, I would like to express my heartfelt appreciation to my family, especially my sisters, husband, and parents, for their unwavering encouragement and support in my educational pursuits.

Table of Contents

Chapter 1: Introduction	1
1.1 Problem Definition.....	5
1.2 Research Objectives.....	8
1.3 Thesis Contribution.....	10
1.4 Dissertation Structure.....	10
Chapter 2: Literature Review.....	12
2.1 Summary	18
Chapter 3: Methodology	19
3.1 Dataset Corpus Collection	20
3.1.1 <i>Speech Sentiment Analysis Classification</i>	22
3.1.2 <i>Text Sentiment Analysis Classification</i>	25
3.2 Speech Dataset Preprocessing.....	26
3.2.1 <i>Speech Dataset Preprocessing</i>	26
3.2.2 <i>Annotation</i>	29
3.2.3 <i>Feature Extraction</i>	31
3.3 Text Dataset Preprocessing.....	33
3.3.1 <i>Datasets Cleaning</i>	33
3.3.2 <i>Word Embedding and Vectorisation</i>	36
3.3.3 <i>Continuous Bag-of-Words Model (CBOW)</i>	38
3.3.4 <i>Skip-Gram Structure</i>	39
3.3.5 <i>Feature Extraction</i>	40
3.3.6 <i>Annotation</i>	41
3.4 Splitting the Dataset.....	42
3.5 Deep Learning Models.....	44
3.5.1 <i>Convolution Neural Network (CNN)</i>	44

3.5.2	<i>The Long Short-Term Memory (LSTM)</i>	47
3.5.3	<i>Bidirectional Long Short-Term Memory BI-LSTM</i>	49
3.5.4	<i>Gated Recurrent Unit (GRU)</i>	51
3.5.5	<i>Generative Pre-trained Transformer (GPT)</i>	53
3.6	Evaluation Metrics	55
3.7	Summary	58
Chapter 4: Implementation and Experiment		60
4.1	Setup and Configuration	60
4.2	Experimentation Construction	63
4.2.1	<i>Collecting and Setting Up the Dataset</i>	63
4.2.2	<i>Applied Deep Learning Models</i>	64
4.3	Findings and Discussions.....	73
4.4	Conclusion and Future Work	93
4.5	Summary	96
References.....		97

List of Tables

Table 1 The Dataset Characteristics	23
Table 2 Dataset Examples and Polarities in the Transcript Speech.....	23
Table 3 Audio files Count.....	24
Table 4 Audio File Size	24
Table 5 The Sentiment Distribution in Text	25
Table 6 The Speech Dataset Labelling and Corresponding Numerical Representation.....	30
Table 7 The File Names Labelling.....	31
Table 8 Confusion Matrix Binary Classification	58
Table 9 The Hyperparameters of the CNN Classifier.....	64
Table 10 The LSTM Hyperparameters	68
Table 11 Hyperparameters for the BI-LSTM Model.....	70
Table 12 Hyperparameters for the GRU Model	72
Table 13 The Performance of CNN on Speech and Text Datasets.....	74
Table 14 LSTM Performance	79
Table 15 BI-LSTM Performance	83
Table 16 GRU Performance.....	87
Table 17 Accuracy in the Models	92

List of Figures

Figure 1 The Framework of the Methodology.....	20
Figure 2 Sentiments in Speech Dataset.....	24
Figure 3 The Dataset Polarities Distribution in the Text Dataset	26
Figure 4 Data Pre-Processing for Arabic Speech Recognition	29
Figure 5 The Signal of Positive Sentiment	32
Figure 6 The Signal of Neutral	32
Figure 7 <i>The Signal of Negative Sentiment</i>	32
Figure 8 The Preprocessing Structure for Text.....	34
Figure 9 Example of the Annotated Dataset	42
Figure 10 The Dataset Split	43
Figure 11 CNN Framework	46
Figure 12 CNN Architecture.....	47
Figure 13 LSTM Memory Block	49
Figure 14 BI-LSTM Architecture	51
Figure 15 The Cell of GRU	51
Figure 16 Example of a Prompt Search (Liu et al., 2023).	55
Figure 17 CNN Performance	74
Figure 18 Confusion matrix for speech SA using CNN	76
Figure 19 Confusion matrix for the text SA using CNN	77
Figure 20 LSTM Performance	80
Figure 22 Confusion matrix for the Text SA using LSTM.....	81
Figure 23 BI-LSTM Performance.....	84

Figure 24 Confusion Matrix for Speech SA Using The BI-LSTM Model	86
Figure 25 Confusion Matrix for the Text SA Using BI-LSTM Model.....	86
Figure 26 GRU Performance	88
Figure 27 The Confusion Matrix for Speech SA Using GRU Model.....	90
Figure 29 Accuracy in the Applied Models	92

Chapter 1: Introduction

Automatic Speech recognition (ASR) is considered an essential instance of digital signal processing, which consists of elements from linguistics, statistics, phonetics, and machine learning. It represents the technology that transcribes spoken words into textual performance, implementing software to interpret human speech and vocal terminologies (Dhouib et al., 2022). Automatic Speech recognition (ASR) is considered an essential instance of digital signal processing, which consists of elements from linguistics, statistics, phonetics and machine learning. It represents the technology that transcribes spoken words into textual performance, implementing software to interpret human speech and vocal terminologies. In recent times, automatic speech recognition (ASR) approaches have gained increasing attention from various speech and linguistic researchers and scholars. This interest is considered evidenced in their incorporation across a variety of domains, for instance, education, robotics, transcription and healthcare (Jaber & Abdulbaqi, 2021). With the rapid improvement in technologies, ASR is now implemented in various applications due to its effectiveness and usability. For instance, employed in transcription software, proposing precious accessibility gains for the users of PC. Another remarkable application includes voice-activated commands and searches on mobile devices. Moreover, ASR systems are implemented for the translation of speech in various languages. As time goes by, ASR approaches have been recognised as a precious input technique among a variety of categories of disabilities, providing alternative services to conventional manual techniques such as mouses and keyboards (Khelifa et al., 2017). On the other hand, ASR is addressed as a challenging issue in the domain of signal processing, demanding several processing layers to achieve the highest accuracy.

Recent technological advancements have greatly enhanced our ability to communicate, access information, and efficiently carry out missions. The introduction of

Web 3.0 has empowered individuals to express their opinions and perspectives on various aspects of their daily lives (S. Cheng, 2024). The availability of individuals' opinions has attracted the interest of businesses, governments, and individuals seeking insights into public sentiment regarding products or services (Li et al., 2014). However, analysing this vast and often disorganised data, which also contains personal content, poses significant challenges. Consequently, fields like big data analytics and sentiment analysis (SA) have become increasingly interested in addressing these challenges (AlKhatib et al., 2020).

Sentiment Analysis (SA) comprises examining individuals' feelings or viewpoints regarding an event, discussions on specific topics, or generally. It finds applications across diverse domains, including understanding human perspectives through their interactions. To enable a machine to grasp human sentiments from conversations, it's essential to identify the speakers and recognise their speech (Dhouib et al., 2022). Therefore, we initially deploy speaker and speech recognition systems before conducting sentiment analysis on the extracted textual data. From the transcribed text, it was possible to extract the sentiment polarities of the sentences and classify them into positive, neutral and negative.

Understanding human emotions can be highly beneficial in various scenarios. For instance, computers equipped with the capability to recognise and respond to non-verbal human cues, such as emotions, could tailor their responses to suit individual preferences and needs.

The prevalence of Arabic audio on social media and other communication platforms in the current digital landscape has highlighted the necessity for sophisticated methods to accurately decipher the sentiment conveyed in Arabic voice. The Arabic language encompasses various accents, including Modern Standard Arabic and Arabic dialects. It is therefore crucial to expand on our knowledge of human influence and to strengthen the effectiveness and resilience of methods used to handle problems with speech sentiment analysis, especially in tasks involving the diagnosis of depression (X. Cheng et al., 2022). In

addition to its use in corporate settings, sentiment analysis (SA) has potential applications in the healthcare industry, particularly in the treatment of mental health disorders. One of the most common mood disorders, depression is characterized by a lingeringly negative impact. Consequently, SA can be quite helpful in recognizing and resolving these kinds of mental health issues (X. Cheng et al., 2022).

The scientific community has studied the endeavored audio files, including news, music, debates, and political conversations. Furthermore, a great deal of work has gone into the analysis of audio data, especially in the areas of multi-speaker discussions and customer service calls (Al-Dujaili and Ebrahimi-Moghadam, 2023). Analyzing audio recordings with multiple speakers can pose challenges. Consequently, the system proposed in this study aims to identify individual speakers and conduct emotion analysis based on their interactions, categorizing emotions into positive, negative, or neutral.

The field of sentiment analysis has seen a rise in popularity recently as a valuable technique for concluding enormous volumes of unprocessed data. Because of the pressing need for these abilities, researchers have developed methods for gauging public opinion in response to this demand for information (Sinha and Kaur, 2016). Businesses and organizations can use the insightful information this analytical technique gives to realign their future goals by gaining a deeper understanding of existing feelings towards a certain product or service. Listening to what customers have to say leads to an improved customer experience. Companies demonstrate their concern for their customers' requirements when they actively seek out and resolve concerns from them. Positive word-of-mouth recommendations are generated, and retention rates are higher as a result of the loyalty and trust this fosters.

Investigation into the optimal utilization of this vast amount of data for emotional analysis continues. Particularly in the field of natural language processing, deep learning, machine learning, and sentiment analysis are seen to be essential to realizing its full potential. By classifying humans' feelings or comments as positive, negative, or neutral based on the indicated attitudes, opinions, or emotions, SA helps organizations gain insightful information. SA has consequently become a potent method for extracting information from massive amounts of unprocessed data. In response to the increasing need for these insights, researchers have concentrated on improving methods for analyzing public opinion (El Barachi et al., 2021). SA helps organizations and institutions make educated decisions and adjust their plans to correspond with current trends in humans' sentiment towards goods and services by giving them access to actionable information.

The importance of SA research cannot be overstated. The increasing number of research on machine learning and sentiment analysis is a testament to its significance. Since the millennium, SA within the realm of natural language processing (NLP) has emerged as one of the most productive areas of research (B. Liu, 2022). However, there remains a lack of resources, tools, and studies focused on languages other than English, particularly Arabic. Additionally, the emphasis has mainly been on English language datasets. Training on big datasets is considered an essential prerequisite for developing robust classifiers, which is a crucial step for various NLP and machine learning applications. For academic research, having access to various small-scale, large-scale and enormous databases is rather advantageous.

Arabic, with 8.2 million native speakers, stands as the sixth most prevalent language globally in terms of speakers ("Arab world - total population 2012-2022," n.d.). Recently, it ranked fourth among the most commonly spoken languages on the websites (Abdelali et al., n.d.). Despite its significant presence, Arabic has seen relatively limited research conducted

in SA compared to other languages. This is relatively due to the complex grammar and rich morphology of the Arabic language, making analysis challenging (Alkhatib & Shaalan, 2017). For this research, an Arabic speech is transcript to an Arabic text as a post, tweet, blog entry, comment, or online opinion. Thus, our objective is to enhance sentiment analysis to accurately distinguish the sentiment tone (positive, negative, or non) of Arabic speech.

This dissertation explores the complexities and methodologies involved in conducting audio sentiment analysis on Arabic Language recordings to investigate the sentiment analysis on the social media data from speech and compare it with the. text data. Subsequently, sentiment analysis is conducted on the speech data attributed to individual speakers, enabling the system to recognise the emotional content conveyed by each speaker, whether it be positive, negative, or neutral. The study presents an Arabic dataset encompassing both Modern Standard Arabic and Dialect Arabic (Alkhatib & Shaalan, 2017). This dataset, collected from YouTube videos, is comparable in size to datasets available for the English language. Given the inherent challenges in analysing the Arabic language due to its intricate structure and rich morphology, the primary objective is to enhance speech sentiment analysis to recognise the sentiment tone (i.e., positive, negative, or neutral) of Arabic speech and compare it with the Arabic textual form. Although the focus is on sentiment analysis applications, the dataset holds potential for various natural language processing (NLP) or machine learning (ML) tasks. The effectiveness of comparison between the Arabic speech and the Arabic text sentiment analyses is evaluated using four classifiers: Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM), and Generative Pre-trained Transformers (GPT).

1.1 Problem Definition

In the recent digital landscape, the widespread presence of the Arabic language on various communication platforms and social media has underscored the need for advanced

techniques to precisely interpret the sentiment expressed in Arabic speech (Dhouib et al., 2022) and the sentiments in Arabic texts. Improving our understanding of how people are influenced and enhancing the efficacy and robustness of approaches to address issues in speech and text sentiment analysis, particularly in missions related to depression identification, has become imperative.

Sentiment analysis (SA) plays a vital role in comprehending human comments and perspectives. Its significance is particularly pronounced in the realm of Arabic speech on social media, where it aids in identifying potential issues, determining client satisfaction levels, and informing strategic business decisions. Additionally, sentiment analysis for Arabic web-based communication faces distinct challenges owing to the unique characteristics of the Arabic language, cultural context, and a lack of large, annotated datasets. Building a robust SA classifier intended for Arabic speech reviews on social media is considered an issue. To obtain valuable insights into customer experiences and perceptions, it is essential for the technique to accurately classify these sentiments as positive, negative, or neutral assessments (Alhumoud et al., 2015). The primary objective is to facilitate online communication, as companies strive to monitor and comprehend customer feedback, identify trends, and make proactive decisions to enhance their strategies.

Apart from its application in the business field, sentiment analysis (SA) can also find utility in the healthcare sector, particularly in addressing mental health issues. Depression, one of the most prevalent mood disorders, is characterized by a persistent negative influence. Therefore, SA can play a significant role in identifying and addressing such mental health challenges (X. Cheng et al., 2022). Identifying human emotions and recognizing depressive states has become a critical concern, essential for both clinical assessments and research investigations.

The following critical challenges are experienced in Arabic Speech Text Sentiment Analysis:

- 1) *The complicated Arabic Morphological*: SA in Arabic dialect make it more difficult than English or alternative languages due to its massive vocabulary, complex grammar, and diversity of dialects.
- 2) *Availability of Arabic Speech Datasets and Text Datasets*: Acquiring a comprehensive and diverse collection of Arabic speech and text datasets from social media opinions is notably limited. The shortage of adequate datasets restricts the development of effective sentiment analysis (SA) systems and the training and evaluation of models. To ensure the robustness of model construction, it is essential to accumulate a significant dataset encompassing a wide range of human sentiments, even positive, negative or neutral and to annotate them accordingly.
- 3) *Polarity Identification of the sentiments*: Identifying the polarity of human emotions demands specialists proficient in sentiment analysis, demanding both time and financial investments.
- 4) *The diversity of Arabic Culture*: Social standards and values significantly impact sentiment analysis for the Arabic language. People's communication styles are shaped by the traditions and customs of diverse cultures. It is crucial to recognize and account for these cultural variations to accurately evaluate emotions and avoid misinterpretations.
- 5) *Homonymy and Polysemy in the Arabic Language*: Arabic expressions exhibit polysemy and homonymy, where the meanings of words may vary depending on the context. This ambiguity makes it challenging to differentiate between positive, negative, and neutral sentiments.

A thorough understanding of Arabic, its cultural context, and the specific characteristics of internet users within the region are essential for effectively navigating these challenges. Furthermore, it underscores the importance of ongoing research and development in the field of Arabic sentiment analysis.

The problem statement revolves around developing a robust sentiment analysis system capable of effectively addressing the challenges associated with evaluating sentiments in Arabic speech and Arabic text and comparing the results with each other. This system aims to provide valuable insights to enhance user satisfaction, improve services, and make data-driven choices to maintain market competitiveness. To achieve this goal, the system will leverage deep learning and natural language processing algorithms.

1.2 Research Objectives

The primary goal of this thesis is to introduce a comprehensive framework specifically designed for evaluating sentiment analysis of Arabic speech and comparing it with the Arabic text conversion. The system aims to address the challenges associated with sentiment analysis in Arabic speech and text and provide dependable sentiment classifications for Arabic social media. The key objectives of the research are as follows:

- 1) *Gather a comprehensive annotated dataset*: Compile an extensive and varied labelled dataset for Arabic speech alongside with annotated Arabic text sentiment analysis, encompassing a wide array of speech and text with the corresponding levels of user satisfaction. These datasets will serve as the primary data source for training and evaluating the performance of the sentiment analysis model in both forms.
- 2) *Constructing deep learning sentiment analysis models assigned to Arabic speech and Arabic text*: Implement deep learning sentiment analysis methodologies that align with the attributes of the Arabic language. Assess strategies for addressing the

complexities of Arabic grammar, cultural norms, and lexicon appropriate to the online social domain, for the audio and text format.

- 3) *Assessing and evaluating Arabic speech and text sentiment analysis models*: Measure evaluation metrics including accuracy, precision, recall, and F1 score. Additionally, generates confusion matrices for the proposed framework using the annotated datasets. To gauge the efficacy of the proposed methodologies, compare the outcomes with those achieved by alternative deep learning techniques for Arabic speech and Arabic text sentiment analysis.

By accomplishing the aforementioned research objectives, this study endeavours to compare sentiment analysis in the Arabic language, either in speech or text forms, particularly within the realm of social media interactions. Gaining a comprehensive understanding of Arabic emotional expressions is crucial for various applications, including human-machine communication in smart cities and psychological healthcare diagnostics. Consequently, there is a pressing need for research initiatives focused on developing robust, culturally sensitive, and linguistically diverse Arabic language SA systems. This endeavour will help bridge the existing gap and contribute to the broader field of sentiment-aware artificial intelligence.

The dissertation targets to investigate the following research questions:

- RQ1) How can deep learning models for Arabic speech sentiment analysis and Arabic text sentiment analysis be effectively categorized?*
- RQ2) Can deep learning models improve sentiment analysis in Arabic speech and text assessments?*

RQ3) What are the most efficient methodologies employed in Arabic speech sentiment analysis and Arabic text sentiment analysis?

RQ4) What is the maximum accuracy attained in sentiment classification for Arabic speech and in sentiment analysis for Arabic text?

RQ5) Which deep learning model yields the best performance when applied to both speech and text formats?

1.3 Thesis Contribution

The primary objective of the dissertation is to address the information gap in Arabic speech and Arabic text sentiment analysis on social media platforms, with a particular focus on both Standard Modern Arabic and dialect Arabic. Various deep-learning techniques for evaluating Arabic speech sentiment analysis and Arabic text sentiment analysis will be introduced and demonstrated. Deep learning classifiers will be employed to achieve superior results in sentiment analysis of social media interactions using the Arabic language.

1.4 Dissertation Structure

- *Chapter 1: Introduction:* proposes a review of applying deep learning models for Arabic speech and Arabic text sentiment analysis on social media platforms comprising the problem definition, study questions, research targets and dissertation participation.
- *Chapter 2: Literature studies* present an extensive review of the literature on Arabic sentiment analysis, including subjects like sentiment polarities among social media platforms. Additionally, research that focuses on Arabic speech sentiment analysis as well as Arabic text sentiment analysis are covered in this chapter. Finally, the essential ideas of this chapter are

briefly summarised in the conclusion.

- *Chapter 3: Background:* review of Arabic speech and Arabic text SA and the fundamental concepts engaged. It investigated several deep learning method designs, the importance of SA recognition, and its difficulties.
- *Chapter 4: Dataset Corpus:* gives a comprehensive explanation of a dataset corpus, covering the procedures for data collection, the utilisation of YouTube as a source of highly emotional content related to Arabic speech and Arabic text, and the annotation process. The Arabic datasets corpus is further presented in this chapter along with a description of the dataset partitioning conduction. A summary of the major subjects covered concludes the chapter.
- *Chapter 5: Methodology:* The strategies employed to achieve the objectives of this thesis are explained in depth in this chapter. This involves applying deep learning methods and combining multiple deep learning strategies on the collected data. A brief overview of the major topics covered concludes the chapter including the evaluation metrics.
- *Chapter 6: Evaluation and Results:* presents the findings and evaluations for the comparison between the algorithms applied to examine the Arabic speech SA and Arabic text model's efficacy. The outcomes are demonstrated in depth and discussed in detail. A summary will conclude the key points covered in this chapter.
- *Chapter 7: Conclusion And Future Work:* This section provides the conclusion for the dissertation. Additionally, it concludes with a concise summary of the directions for future research.

Chapter 2: Literature Review

This chapter covers several topics, such as the usage of sentiment analysis on social networks and its application, as well as an understanding of natural language processing (NLP) with a focus on Arabic speech SA works. After that, it examines more sentiment analysis techniques, such as deep learning and machine learning. After that, the extracted features and data pretreatment techniques will be examined.

About 95% of big data is available as unstructured text that is gathered from websites, email messages, and other comparable resources. New scientific techniques and instruments are required to completely reap the benefits of this enormous amount of data (Cai, 2021). This endeavour has been made easier by Natural Language Processing (NLP) tools, which process enormous amounts of textual data effectively. NLP is a branch of artificial intelligence that uses computational techniques such as deep learning as well as machine learning to understand, learn from, and produce coherent language that is comparable to humans (Cai, 2021). Studies like the one carried out (Guetterman et al., 2018) show that NLP can produce results that are on par with those obtained by classic qualitative textual analysis techniques. This study showed that while natural language processing (NLP) was able to identify significant themes that were discovered by traditional text analysis, it was less successful at capturing nuances and providing contextual details. NLP has developed in a wide range of industries and has applications in online communities, medical settings, medical care, and other areas. Although methods based on deep learning have been extensively incorporated into natural language processing (NLP), their use in clinical settings is still restricted because of their inferior interpretability in comparison to conventional algorithms for machine learning (Casey et al., 2021). Sentiment analysis saw a rise in the use of syntax-based approaches, a subset of natural language processing (NLP) strategies,

especially in the years 2016–2017. However, as sentiment analysis has progressed, traditional machine learning classifiers and deep learning-based models have replaced pure natural language processing techniques. These new approaches have produced cutting-edge outcomes and shown superior accuracy (Khairi et al., 2020). All the studies in the field of Arabic NLP have only scratched the surface, despite the Arabic language being a treasure trove of fascinating topics to explore.

In the last decade, Arabic and its dialects have gained interest in the field of Natural Language Processing (NLP) research. Various studies have explored different aspects of processing this language and its dialects, including the analysis of the morphological features, the translation process through automation, and the development of the resources. Several studies have been proposed to categorise Arabic-language works and clarify their linguistic characteristics (Alkhatib & Shaalan, 2017); (Dukes & Habash, n.d.) and (Shoufan & Alameri, 2015). There is still a need for further research in the field of Arabic NLP, as existing studies have not yet yielded comprehensive results on this topic.

In recent years, sentiment analysis has undergone significant exploration, particularly concerning textual data from social media platforms. Previous techniques, primarily lexicon-based, have evolved towards statistical and deep learning-based models. Particularly, SentiStrength (Thelwall et al., 2010) is a commonly identified lexicon-based technique, leveraging frequently utilised sentences and terms on social networking platforms to analyse short textual. based on this, Saif et al. proposed SentiCircles (Saif et al., 2016), a method that specified Twitter’s sentiment analysis. By accounting for express co-occurrences among varieties of tweet comments, SentiCircles proposed improved analytical features. Furthermore, enhancements in deep learning have simplified the application of sequential techniques like Long Short-Term Memory (LSTM) (Huang et al., 2016) and convolutional

neural networks (CNNs) (Alayba et al., 2018; Y. Kim, 2014) for feelings recognition in comments. (Shin et al., 2017) shows a hybrid algorithm by integrating CNNs with lexicons utilising attention techniques, additionally improving sentiment analysis function. In addition to textual data, the proliferation of speech and video content on social website platforms has attracted concern in visual sentiment analysis. Two primary methods emerge in this area: deep learning and mid-level representations. Specifically, Borth et al. improved SentiBank (Borth et al., 2013); (Borth et al., 2013), a library able to recognise visible concepts up to 1200 pairs of adjective-noun (ANPs) for sentiment identifications. Meanwhile, minimum visual cues, such as the "gist" models (Oliva & Torralba, n.d.) and wavelet-like Haar features (Viola & Jones, 2001), have also been applied. (Yuan et al., 2013) investigated sentiment estimation in images by face detection and benefiting from facial expressions alongside low-limit data of visual. The downside to these studies is the necessity for robust language resources, which may not always be readily available.

Millions of people who live in countries where Arabic is considered the home language have a relationship with Islam. Arabic is a Germanic language consisting of 28 letters. According to (*Arabic Speaking Countries List - 2023 Update*, 2023), there are more than 456 million original Arabic speakers in the world as of 2022, distributed all across 22 Arab-speaking countries, or roughly 5% of the total population. It is estimated that by 2030, there will be about 581 million Arabic-speaking people on the planet, while the number of the population who speaks Arabic will be around 724 million by 2050 (*Middle East Demographics to 2030*, 2019). Based on that, Arabic has seen a remarkable increment in popularity and is recently rated as the fourth most regularly spoken language on the internet (Abdelali et al., n.d.). Nevertheless, the amount of studies on natural language processing corresponding to Arabic may not be considered with the language's extensive utilized (Mozetič et al., 2018). In addition, sentiment analysis strategies in Arabic have not improved

to the same level as those in other languages such as English. However, scholars such as those performed by (Alkhatib & Shaalan, 2017) demonstrated that the number of studies dedicated to Arabic emotion analysis is in increments. The investigators (Oussous et al., 2020) applied and assessed two deep learning approaches: long short-term memory (LSTM) and convolutional neural network (CNN). For Arabic sentiment analysis (ASA), the structure provides several preprocessing methods. The study initially released the Moroccan Sentiment Analysis Corpus (MSAC), a new, extensive Arabic corpus that is accessible to the public because of this effort. Second, the suggested framework showed how ASA has improved. The experimental findings demonstrated that deep learning models outperform classical methods (support vector machines, naive Bayes classifiers, and highest entropy) in terms of ASA efficiency. Moreover, the research demonstrated the critical significance that morphological characteristics play in Arabic NLP. The complexity of sentiment analysis in Arabic, which includes not just Modern Standard Arabic (MSA) but also several dialects spoken in various places, is another factor contributing to the scholarly divide. Sentiment analysis requires specialised methodologies since different dialects may display distinct linguistic traits and emotion analysis.

The research by (Mourad & Darwish, 2013) presented a manually annotated corpus in which newswire piece sentences were individually annotated. The investigators used this dataset, which consists of 2,855 phrases, to demonstrate an automated method for subjectivity and sentiment analysis. Utilising the F measurement criteria to evaluate the findings, the researchers investigated how various preprocessing framework grades affected the mission of categorising subjectivity and emotion analysis. They showed that the combination of stemming, morphological and adjective qualities produced a maximum F-measure of 71.54% in terms of subjectivity classification. The combination of stemming with morphological and adjective features, however, received the greatest score of 95.52% for the categorisation of

emotions. (A. Aladeemy et al., 2024) utilized two different datasets named ASTC and Artwitter, which are related to social media platforms employed in this investigation. The availability of Arabic corpus sources remains limited, constrained by specific annotated data, necessitating further investigation in this domain.

Many studies have turned to hybrid techniques in sentiment analysis since machine learning algorithms require huge, labelled datasets for training. These techniques blend machine-learning training with polarity derived from lexicon-based methodologies. High precision but low recall was observed in lexical techniques (Keith et al., 2017), which led them to hybridise MPQA with SVM for improved performance. (Mendon et al., 2021) found that when lexicon is integrated into hybrid techniques, the neutral class classification problem is addressed, leading to overall accuracy gains when tested using NB, LR, SVM, and DT classifiers across product, Twitter, and movie datasets. The researchers also underlined the importance of feature extraction and discovered that TF-IDF produced the best accuracy.

The lack of investigation and creation of efficient methods designed especially for the Arabic language is the cause of the study gap in Arabic sentiment analysis. Though sentiment analysis is becoming more popular in a number of languages, including Chinese, Russian, and English, studies and methodologies specifically for Arabic are conspicuously lacking. The lack of well-labelled datasets created especially for Arabic sentiment analysis is mostly to blame for this discrepancy. While there are materials available, they are frequently of limited scope and might not adequately convey the variety of linguistic subtleties and social circumstances that are inherent in Arabic speech.

Systems for identifying emotions in conversation vary depending on the language, models used, and features collected from datasets. Not much study has been done, especially on Arabic speech, even though many studies have tried to extract emotions from voices in

different languages. The investigators (Alamri & Alshanbari, 2023) employed the KSUEmotions corpus (Meftah et al., 2021), which is recognized as one of the most substantial Arabic corpora, for Arabic SA. Different models were used in the study to identify sentiments. The study implemented Support Vector Machine (SVM) and K-Nearest Neighbor (KNN), in addition to deep neural networks (DNN), convolutional neural networks (CNN) and bi-directional long short-term memory (BLSTM). The sentiment evaluation and emotion recognition approaches were used in a different research project to identify feelings in Arabic spoken words (Hussein et al., 2022). In this work, the authors utilized the Arabic language and its dialects via human speech recognition (HSR), HMM–DNN ASR modular, and end-to-end transformer ASR. The first End-to-End (E2E) model for Arabic Automatic Speech Recognition (ASR) was presented by Ahmed et al. (2018). It used recurrent neural networks in conjunction with Connectionist Temporal Classification (CTC). This basis has been built upon by other developments, such as models based on transformers that demonstrate fluency in Modern Standard Arabic (MSA) and dialects (Hussein et al., 2021). The requirement for annotated data is a major obstacle for E2E ASR models, especially in languages with limited resources like different Arabic dialects. Self-supervised and semi-supervised learning techniques are becoming more and more popular as a means of addressing this problem. These models, like Wav2vec2.0 along with XLS-R, are capable of being fine-tuned for specific functions after first learning useful representations from large volumes of unlabeled or badly labelled data (Babu et al., 2021). W2v-BERT, a different kind of self-supervised method, uses contrastive learning strategies with masked linguistic modelling. It has been used for Arabic ASR by fine-tuning on the FLEURS dataset, which represents the Arabic Egyptian dialect. Wav2vec2.0 and w2v BERT, in contrast to Whisper, need a fine-tuning stage to properly decode data.

A speech spectrum is a valuable resource since it contains a vast amount of information. It can be difficult to identify the best way to gather sentiment-related data. Strategies for users need to be efficient with resources, which aligns with the challenge of finding the right set of emotion-related data. The shortage of corpora limits the wider application of machine learning techniques, especially when deep learning algorithms are used, which limits the range of previous efforts. These latter strategies have proven to be remarkably effective, as recent developments in sentiment analysis have shown.

The research gap exists in the restricted investigation and comparative inspection between Arabic audio and textual sentiment analysis. Despite the rising concern in sentiment analysis in both systems, there is a remarkable lack of research that systematically studies and compares the productivity, robustness, and deficiencies of Arabic emotions analysis in speech and text forms. This gap restricts an extensive knowledge of the efficiency of sentiment analysis methods among various lingual Arabic modalities and limits the enhancement of developed approaches assigned to applications and frameworks.

There is a scarcity of studies comparing Arabic speech sentiment analysis with Arabic text sentiment analysis to determine which one performs better. Therefore, the objective of this study is to conduct such a comparison.

2.1 Summary

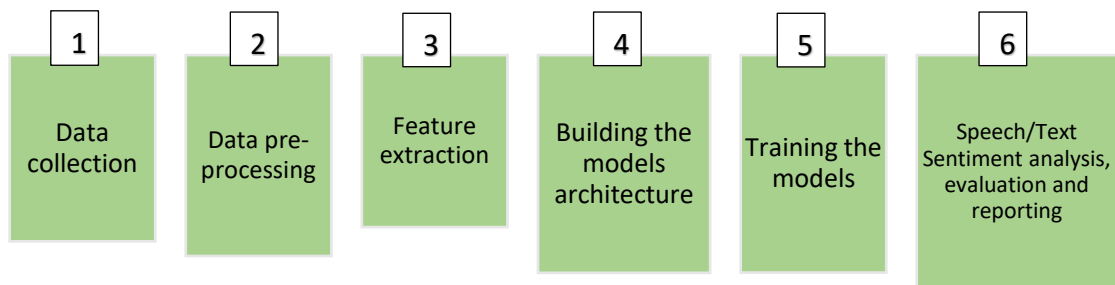
This chapter provides a summary of recent and fundamental research on sentiment analysis, with particular attention to sentiment analysis in Arabic. An overview of NLP for the languages Arabic and English is presented. The sentiment analysis in Arabic, sentiment analysis techniques, and sentiment analysis on social networking sites are also covered in this chapter. Additionally, it highlights voice sentiment analysis, specifically for Arabic.

Chapter 3: Methodology

Deep learning mechanisms are utilised to examine the sentiment terminologies in dialectal expression usage to understand the sentiment's nuance contained within expressions. This aspect of artificial intelligence improves man-machine interactions by giving AI systems the ability to understand and react to emotional speech and messages in language. Arabic is a difficult language to learn because of its many dialects, distinctive phonemes, and cultural nuances. As a result, emotional recognition demands particular mechanisms. The Arabic sentiment analysis task in the suggested technique only uses a neural network. This system's main goal is to determine the sentiment situation that is expressed in people's comments. The neural network then goes about mining features from the input data so that it can be easier to identify the subtle emotional overtones in the text.

This method, which focuses on dialectal words, even written or spoken, describes the meticulous procedure for creating a trustworthy SA system designed for Arabic people. The process is designed to take into account the linguistic complexity and emotional nuances unique to Arabic at every step, from data collection to model training and application. The introduced system's framework is shown in Figure 1. Below is the methodology structure for this dissertation.

1. Gathering of Data
2. Preprocessing the dataset
3. Extractions from feature engineering
4. Creating the Model's Structure and Educating It
5. Assessing and Adjusting the hyperparameters of the model
6. Evaluation of Model functionality

Figure 1*The Framework of the Methodology*

3.1 Dataset Corpus Collection

Consistent with this (Alshutayri & Atwell, 2019), there is a merging of Arabic dialects on social networking platforms, such as (DA) and (MSA). These language variants reveal notable distinctions that can affect the application of (SA), while they also have certain commonalities. Arabic speakers use dialect Arabic, also defined as "colloquial Arabic," in daily discussions. This accent is widely used throughout many nations and areas. Included in this category are conversed Arabic accents from the Gulf, the Maghreb, Egypt, and a few others. Dialect Arabic is extensively used in writing, with regional differences, by users of online platforms and the internet (Sadat et al., 2014). This thesis is mostly concerned with the Arabic dialect and Modern Standard Arabic (MSA).

Arabic uses both temporal and affixational morphemes, has a complex component structure, and several morphological rules. Arabic makes extensive use of pronouns and related phrases, which are found in a wide variety of inflexions. Arabic's intricate morphology is made even more complex by the more than 10,000 roots that serve as the basis for nouns and verbs (Shaaan et al., 2018). The 120 patterns that make up Arabic morphology emphasise the importance of the 5,000 fundamental roots. Arabic also has a

variable word order, meaning that words can be arranged in a sentence to highlight particular parts. In order to ascertain sentence structure that conforms to Arabic grammar rules, the syntactic analyser parses the input tokens supplied by the dictionary analyser (Alkhatib and Shaalan, 2017).

This section delves into a comprehensive discussion of recommended data collection techniques. Arabic social media platforms encompass a spectrum of linguistic varieties, including (MSA), (DA), and Blended Arabic (BA). While these varieties share some similarities, they also exhibit distinct differences that may impact the efficacy of sentiment analysis (SA). Particularly in the realm of machine learning, Python and its libraries stand out as widely adopted and versatile tools for data analytics. Various methods exist for corpus labelling; traditional approaches entail manual labour, while computer-aided techniques leverage annotation devices, among others. This data is primarily gathered from multiple websites, including YouTube, Facebook and Twitter.

To identify specific scenes that exhibited blatant emotional cues, the evaluation group closely watched and assessed each video. The study aimed to assess how well various machine learning algorithms identified sentiments. The audio recordings were then separated into smaller chunks after the video was processed to extract them. Every part was recorded in MP3 format, with a duration varying between one and three seconds. The natural pauses in speech were used to segment the speech, ensuring that every segment contained a whole phrase. The goal of sentiment analysis techniques is to assign texts to these particular categories, providing important information about the text's emotional background. These categories aid in the understanding of feedback, public opinion, and trends in various textual. The sections below propose samples of the transcribed Arabic speech into Arabic text. Following the preprocessing phase and the elimination of extraneous elements such as music,

silence, and background noise, a comprehensive analysis of sentiments and their respective intensities is extensively discussed. As a result, these feelings served as labels for the audio clips. Before classification, as part of the data preparation phase, steps were taken to eliminate any undesired interference, such as background noise, quiet periods, and music.

3.1.1 Speech Sentiment Analysis Classification

An array of emotive speech in the Arabic dialect was collected for this dissertation from videos on YouTube, Facebook, and Twitter. They include a variety of programs, including comedy performances, scenes, and short films. Data were merged to form a dataset reflecting reviews categorised into positive, negative, and neutral sentiments. From June 2022 to August 2023, approximately 300 speech comments and suggestions were compiled, which included a corpus of more than 4.22 GB of data. About 40 Arabic utterances with an Arabic dialect are included in the sample. Over two 20 male and approximately 20 female Arabic speakers, ages 23 to 75, collectively provide more than 8 hours of speech to these assertions. Table 1 outlines the attributes of this classification. The evaluations were segregated into three categories: positive (POS), negative (NEG), and neutral (NON). The speech is then transcribed into text to study. The two CSV files were generated to accommodate both textual and verbal feedback. Given CSV's compatibility with multiple text processing applications, the review content was initially extracted into an Excel file and subsequently converted into CSV format for preprocessing purposes.

Table 1*The Dataset Characteristics*

Period	Date
Start Date	June 2022
End Date	August 2023

Table 2*Dataset Examples and Polarities in the Transcript Speech*

Comments	Translate into English	Category	Labels
انا سعيدة واتمتع بصحة جيدة بسبب هذا المنتج	I am healthy because of this product	Positive	POS
لا يجب عليك التصرف بهذا الشكل مع الاطفال	You shouldn't act like this with children	Negative	NEG
تعد ممارسة الرياضة من السلوكيات المفيدة للصحة	Exercising is a behaviour that is beneficial to health	Neutral	NON

Table 3 and Figure 2 present data related to Arabic speech sentiment analysis, particularly focusing on the distribution of sentiments within a dataset consisting of audio files. It categorises the sentiments into three main categories: Positive, Negative, and Neutral. The corresponding number of audio files accompanies each sentiment category. Specifically, there are 2810 audio files labelled as Positive, 2963 as Negative, and 2675 as Neutral. The total number of audio files in the dataset is 8448.

Table 3*Audio files Count*

Datasets	Sentiments			Total
	Positive	Negative	Neutral	
Audio files count	2810	2963	2675	8448

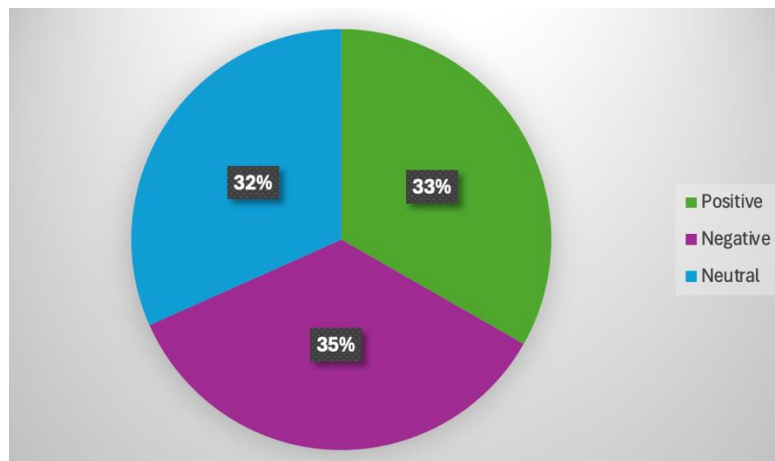
Figure 2*Sentiments in Speech Dataset*

Table 4 presents data about collected sentiment analysis datasets, focusing on the size distribution of audio files. It categorises the sentiments into three distinct classes: Positive, Negative, and Neutral. The corresponding size of the audio files accompanies each sentiment classification. Specifically, the Positive sentiment category comprises audio files of 1.32 GB in size, while the Negative sentiment category includes 1.57 GB of audio data. Similarly, the Neutral sentiment category contributes 1.42 GB of audio content. Overall, the total size of all audio files across all sentiment categories is about 4.22 GB.

Table 4*Audio File Size*

Datasets	Sentiments	Total
----------	------------	-------

	Positive	Negative	Neutral	
Audio File Size	GB 1.32	GB 1.57	GB 1.42	4.22 GB

3.1.2 Text Sentiment Analysis Classification

The transcribed dataset, which shows the classifications of SA, was produced by compiling reviews from social media platforms. The total number of comments and feedback received between June 2022 to August 2023, as displayed in Table 5, was about 8677 comments. Since CSV is compatible with several text processing programs, the review material was first extracted into an Excel file and then converted to a CSV format for preprocessing. Table 5 provides a breakdown of sentiment analysis datasets, specifically focusing on the distribution of sentiments within textual data. It categorises the sentiments into three main classifications: Positive, Negative, and Neutral. The corresponding count of text entries accompanies each sentiment category. Specifically, there are 2853 instances of text labelled as Positive, 2903 as Negative, and 2921 as Neutral. The total count of text entries in the dataset amounts to 8677. This table offers valuable insights into the distribution of sentiment-labelled textual data, facilitating a better understanding of the sentiment composition within the dataset.

Table 5

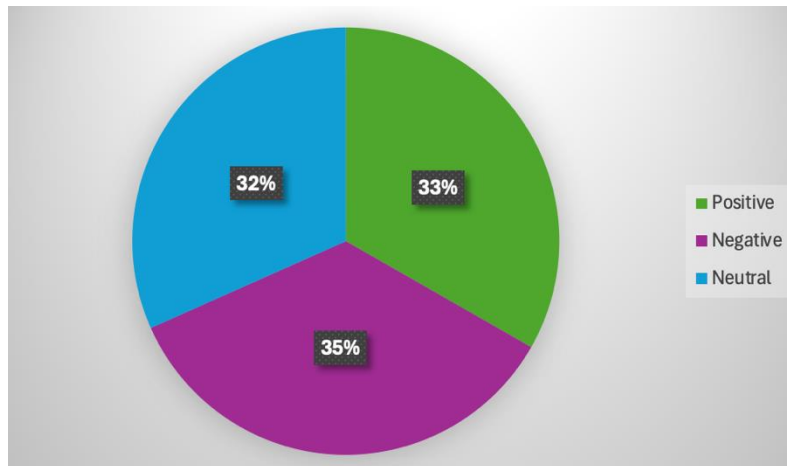
The Sentiment Distribution in Text

Datasets	Sentiments			Total
	Positive	Negative	Neutral	
Text count	2853	2903	2921	8677

Figure 3 illustrates the distribution percentages of the text datasets across positive, negative, and neutral sentiments. The graph indicates a nearly equal distribution among all categories, with a slight increase observed in the negative class.

Figure 3

The Dataset Polarities Distribution in the Text Dataset



3.2 Speech Dataset Preprocessing

In order to improve the quality of the obtained dataset and guarantee increased efficacy in sentiment analysis, pre-processing is an essential step in both SA and NLP. Text cleaning, normalisation, tokenisation, stop word elimination, and stemming are among its main goals. The pre-processing process could involve several processes, based on the goals of the study and the language that was selected. One-of-a-kind pre-processing was used in research evaluating Arabic sentiment categorisation effectiveness. This required using a personalised stop list and emotional support as tactics. According to the results of the experiment, Arabic emotion categorisation can be improved by substituting bespoke stop lists and content words for emojis.

3.2.1 Speech Dataset Preprocessing

Preprocessing, the early grade preceding subsequent phases of the speech recognition operation, stands as the prime step in developing an ASR system. This is mainly done to

discriminate signals that are generated with vocal cord vibrations from those that aren't and to build vectors that extract pertinent features. Preprocessing is done on the voice signal to make it more suitable for feature engineering extraction analysis later on.

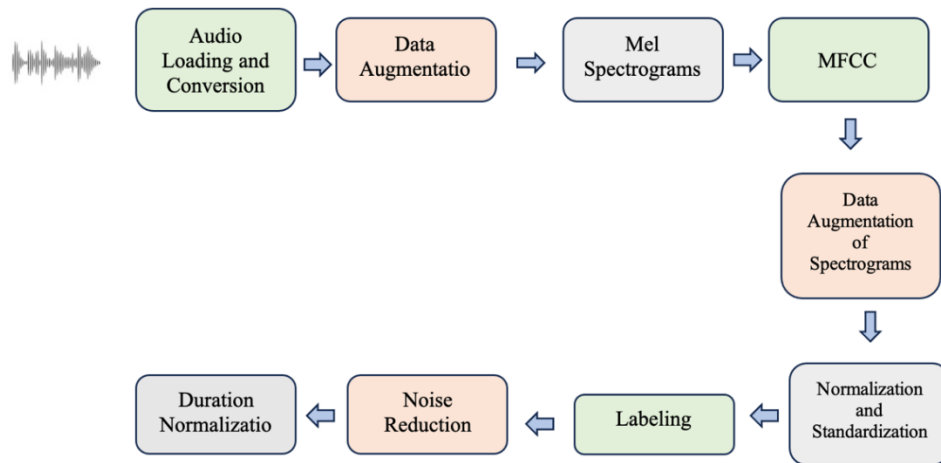
The target of data preparation in ASR is to develop the feature of received voice data and streamline the operation of extracting valuable characteristics. To ensure that the acoustic signals are appropriately prepared for additional analysis by ASR models, several steps are involved in the process. The following are crucial elements in preparing data for Arabic speech recognition, as summarised in Figure 4:

1. **Audio Importing and Conversion:** The first step is to import audio files in the commonly used .wav format for speech data that comprise spoken utterances. After the audio files are parsed, the waveform is converted into a 2D NumPy array, which shows the sound's amplitude or strength over time.
2. **Augmentation of the dataset:** To develop the dataset's variation and circularity, augmentation mechanisms are implemented. This could entail slightly altering the pitch or shifting the audio's spatial placement to the left or right. The ASR algorithm can learn from a larger range of input variables with the help of augmentation.
3. **Mel Spectrograms:** Mel spectrograms are produced by transforming the audio input. Spectrograms use a breakdown of the audio signal into a set of frequencies across time to visually represent the signal. The frequency components of the speech signal are precisely captured by mel spectrograms, and this is important information to extract features for Automatic Speech Recognition (ASR).
4. **Mel-Frequency Cepstral Coefficients (MFCC):** The voice signal undergoes a transformation that yields MFCC in addition to spectrograms. Compact and

essential, MFCCs are a set of coefficients extracted from the audio spectrum.

When it comes to accurately capturing the essential elements of human speech, MFCCs excel.

5. **Augmenting Spectrogram Data:** Relying only on spectrograms while applying data augmentation techniques improves the dataset even further. Techniques like temporal and frequency masking, as in SpecAugment, are often used to improve the robustness of models.
6. **Normalisation and Standardisation:** To ensure that the entry characteristics have uniform scales, normalisation processes are applied. For the model to converge during the training process, this is necessary. The data can be adjusted to have a mean using standardisation, which normalises the dataset.
7. **Classification:** To facilitate supervised learning, each audio clip is assigned to the relevant emotion.
8. **Noise Reduction:** Preprocessing may involve removing spurious noises, calm patches, or background noise from the audio file. Techniques for reducing noise improve the signal-to-noise ratio, which improves the model's ability to recognise speech.
9. **Duration Normalization:** Audio files may have different durations; therefore, in order to facilitate effective learning from the data, the durations must be normalized to a consistent length. A common tactic is to average the duration of each audio recording in order to prepare them for modelling. The input data ASR is enhanced by employing these preprocessing techniques, which makes it possible for ASR models to effectively gather and recognize speech patterns in Arabic language utterances.

Figure 4*Data Pre-Processing for Arabic Speech Recognition***3.2.2 Annotation**

Since ML and DL mechanisms mostly rely on annotated data to find styles and create predictions or classifications on new, unknown data, annotation is crucial. Models might not be able to operate to the best of their abilities without accurate and comprehensively annotated training data, which could lead to inaccurate results.

In the field of emotional speech recognition, annotation is essential due to it being a first step towards training DL and ML mechanisms. SA using emotion identification models finds relationships and patterns in speech signals that are indicative of different emotional states mostly by using labelled data. These models learn and use their understanding of various sentiment cues based on annotated datasets.

Annotators annotate voice data segments with labels as positive, negative and neutral to create a trustworthy reference for model training. The level of accuracy and comprehensiveness of these labelling directly impacts the model's ability to accurately identify and classify speech emotions. The model's performance is enhanced by fully

annotated datasets, which ensure robust and consistent emotion identification abilities when faced with unknown speech samples.

After the dataset was gathered, the initial task was to recognize and comprehend the voice tapes. Every dataset had a unique naming scheme, and the file names were used to determine the emotion label for subsequent classification. Table 6 shows the labels and the corresponding numerical representations. Table 7 presents how to label the filenames in an example manner. Wave charts, which showed randomly selected audio samples, were used in the dataset analysis. These plots were intended to provide a visual representation of the properties of the data under investigation. The quality of the speech data was also assessed by looking for any potential surrounding noise in the voice tapes and gauging how well participants understood the emotions that were being conveyed.

Table 6

The Speech Dataset Labelling and Corresponding Numerical Representation

Sentiment Type	Labels	Numerical Representation
Positive	POS	0
Negative	NEG	1
Neutral	NON	2

In this dissertation, annotation indicates the procedure of labelling or marking each audio segment with certain emotion categories. The audio files were given names based on the file ID name and the acronyms for the diverse sentiment categories. The first two alphabet characters that signify the emotion type are extracted by the MFCC_Data_PreProcess function to obtain the labels from the filenames. After extraction, the filenames are used as the label. The labels are added to the data dictionary's "labels" list by the audio segments' emotional content.

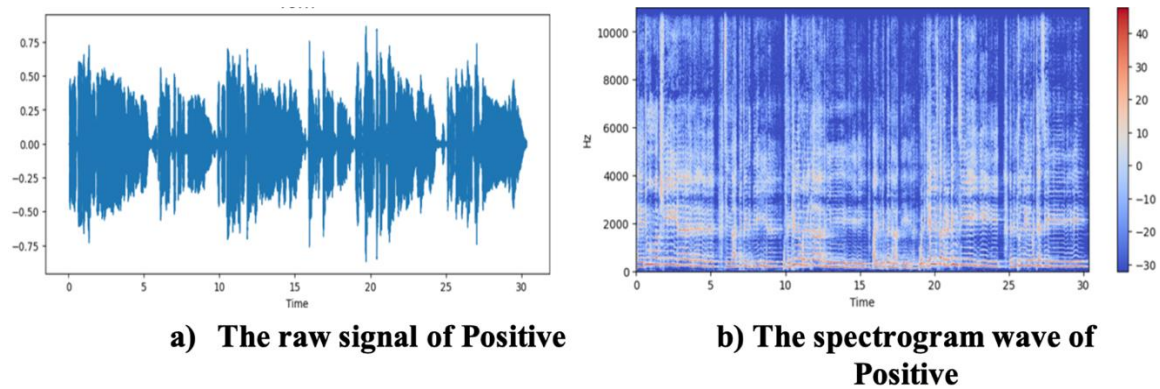
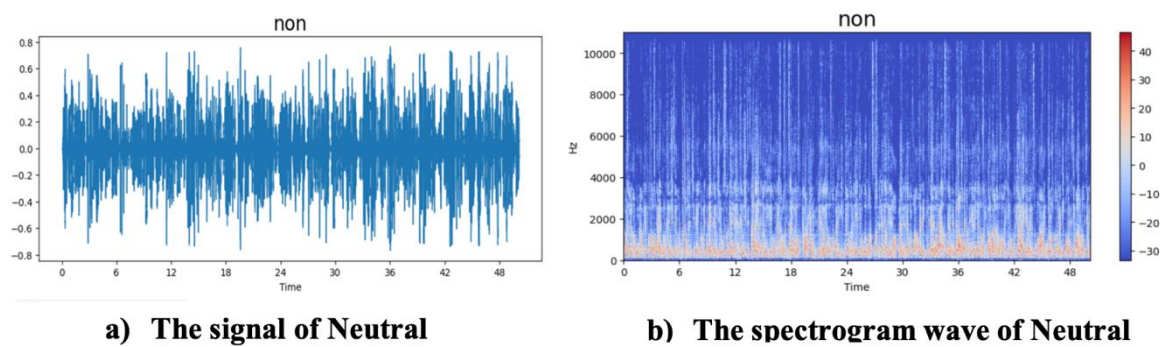
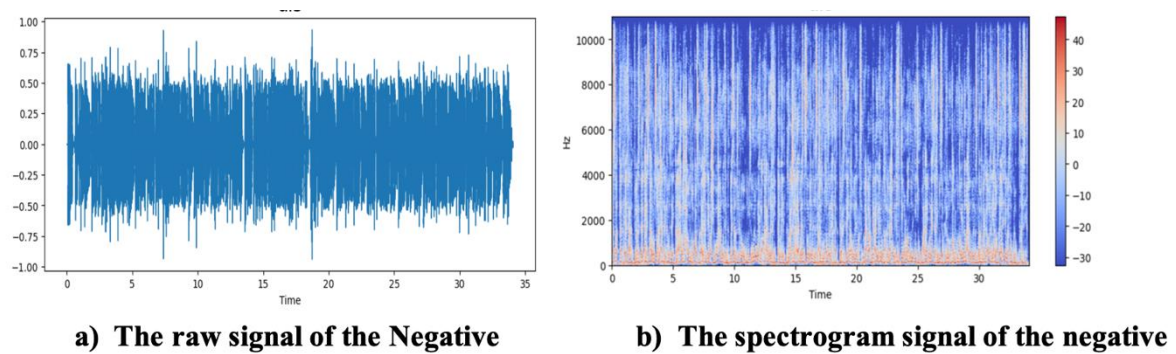
Table 7*The File Names Labelling*

FileNames
_POS.wav سعيد
_NEG.wav زعلان
_NON.wav صحة

3.2.3 Feature Extraction

The voice signal is made up of several components that convey different emotional qualities. Mel-Frequency Cepstral Coefficients, or MFCC for short, is a technique that gathers crucial data from voice data segments to identify significant features for voice analysis. An audio signal's Mel-frequency cepstral coefficients (MFCCs) are a short set of features that represent the spectral envelope pattern; these features typically range from 10 to 20. Logarithmically spaced frequency filters are used for higher frequencies and linearly spaced frequency filters for lower frequencies in the computation of MFCC characteristics. As per the findings of (Saxena et al., 2022), this design replicates the low-frequency sensitivity observed in human ears. The MFCC matrix computation for each audio sample is shown in the following images.

The MFCC matrix calculation for each audio sample is shown in the following images. They display the emotions' unprocessed raw signals as well as the MFCC features visualized using spectrograms for every emotion. The signals for the positive raw data and the spectrogram wave are shown in Figure 5 (a) and (b), respectively. Two signals are shown in Figure 6: The raw data signal for the neutral sentiment (a), and the negative spectrogram wave signal shown in Figure 5 (b). For negative, figure 7 shows the raw data and the spectrograms wave.

Figure 5*The Signal of Positive Sentiment***Figure 6***The Signal of Neutral***Figure 7***The Signal of Negative Sentiment*

3.3 Text Dataset Preprocessing

To improve the quality of the obtained dataset and guarantee increased efficacy in (SA), pre-processing is an essential step in both (SA) and (NLP). Text cleaning, normalisation, tokenisation, stop word elimination, and stemming are among its main goals. The pre-processing process could involve several processes, based on the goals of the study and the language that was selected. One-of-a-kind pre-processing was used in research evaluating Arabic sentiment categorisation effectiveness. This required using a personalised stop list and emotional support as tactics. According to the results of the experiment, Arabic sentiment categorisation can be improved by substituting bespoke stop lists and content words for emojis.

3.3.1 *Datasets Cleaning*

Data cleansing is an essential first step towards improving data quality. This method includes deleting slang and errors after data collection. There are difficulties when adding this to Arabic writing because of the wide range of Arabic dialects and linguistic variances. In the field of Natural Language Processing (NLP), Arabic has distinct challenges because of its multilingualism and intricate morphology, requiring a great deal of pre-processing. Eliminating hashtags, URLs, punctuation, and superfluous whitespace are effective way to sanitise online social media corpora. Furthermore, depending on the particular work at hand, people may decide to eliminate particular symbols, digits, long words, and English language material. To make sure that raw Arabic content is refined and appropriate for analysis, it is important to evaluate and preprocess it.

Remarkably, there are few publicly available datasets for researching Arabic-language internet product reviews and discussion board customer service ratings. This lack of availability is probably due to the difficulties in gathering these datasets under the strict

guidelines set by social media companies. As such, information from Shein.com and Amazon.com was used to create databases specifically for this research. In this section, a range of data preparation and cleaning methods was used to reduce the Arabic language's complexity and improve the quality of the results. Throughout the data preprocessing stage, Python was used. The structure of data preprocessing and cleaning is shown in Figure 8, along with an outline of the next steps. Preprocessing and cleaning the dataset are critical before doing sentiment analysis. Preprocessing and cleaning the data are crucial before doing sentiment analysis. Preparing and normalising the dataset is the initial stage in preprocessing, which guarantees that the classification technique can yield quick and correct results. Figure 8 summarises the preprocessing steps.

Figure 8

The Preprocessing Structure for Text



The following are the steps that are involved in this step:

1) Reading and Uploading Data: The gathered dataset was loaded using the standard Python module, and the data frame was read to perform preprocessing operations.

2) Diminished Noise:

- Removing any reviews or comments that are not in Arabic.
- Eliminating any Arabic or English numerals, such as (١, ٢ or 1, 2).
- Eliminating all punctuation, including (،, ., ؟).
- Eliminate URLs, emojis, expressive stickers, hashtags, and hyperlinks.
- Taking out emojis such as (٧).

- Eliminating quotation marks and tatweel ("!, ~").
- Eliminating double-backed characters such as (كيسير).

3) Transformation: Changes comprise repositioning the beginning and ending punctuation, URLs, hashtags (#), mentions (@), lowering the case of capital letters, and swapping out several spaces for a single space.

4) Tokenisation: Tokenization is a technique used in machine learning and Natural Language Processing (NLP) that breaks down a text sequence into smaller units called tokens. The size of these tokens might range from single characters to whole words. This technique is important because it breaks down human language into smaller, easier-to-understand parts, which makes machine comprehension easier.

5) Filtering (Stop word removal): This stage entails getting rid of words like pronouns, prepositions, and articles that don't matter for text mining. Stop phrases such as "in, في", "which, اي", "before, قبل", and other often-used expressions have no semantic significance to the context of the text. They can be safely excluded without affecting the performance of the classifier because they are frequently unnecessary for the categorization task (Makrehchi and Kamel, 2017). The procedure is improved overall when there is less noise. Since our goal is to improve the SA system's accuracy and efficiency, we decided to eliminate stop words before using any categorization techniques.

6) Lemmatization: The system of decreasing concepts to their fundamental form, or lemma, is known as lemmatization in NLP. The lemma represents a word's dictionary form, which is linguistically acceptable and can stand alone as an original term.

7) Remove any irrelevant instances: Look for and eliminate any symbols that have no bearing on the current text. If the goal is to classify emotion, then, for example, leave out examples that don't express emotion.

8) Stemming: In opinion mining and natural language processing, replacing a word with its stem is one of the most crucial pre-processing procedures (Mustafa et al., 2017). Stemming standardises language by reducing all words to their most basic form. The stem of many Arabic words is the same. The Arabic terms (علم- معلم- معلوم- متعلم), for instance, all share the same stem (علم), therefore, eventually the phrase (كتب) will take the place of each of these words. This method significantly improves the effectiveness of text categorisation through phrase removal, corpus reduction, and other optimisations. Root extraction and mild stemming are two different kinds of stemming techniques. Light stemmers look for and remove infix patterns to determine the root of a word. Word length and stemming phases are sometimes used as criteria. Light stemmers can perform better if pertinent suffix and prefix lists are created and word length requirements are met (Alkhatib et al., 2017). However, dissecting a conflation into its root allows for root extraction, one of the main text-processing methods. This process aims to resolve the morphological complexity problem with the Arabic language (Khafajeh et al., 2018).

3.3.2 Word Embedding and Vectorisation

In NLP, a process called word embedding converts words into numerical representations that are comprehensible to machines. For text representations that have been taught as numerical representations, this is a typical method. Each term must have a number structure allocated to it, since the technique requires assistance in understanding term handling (Salama et al., 2018). You can accomplish this in several ways:

1. One-hot encoding: a text's words are given distinct numbers. This value is converted into a binary vector that represents the word by utilising 0s and 1s.
2. A count-based system, in that the count of occasions a word shows in the text body determines its number.
3. SLIM combination: Using both of these methods, the SLIM combination helps a computer understand both the meaning and frequency of terms in the text.

Word embedding functions within the field of (NLP) by displaying words as closely spaced arrays of exact numerical values in up to 1000 dimensions. Vectorisation is the process by which words are transformed into numerical vectors. A sparse vector, which has a large number of zero elements and is suggestive of one-hot encoding, is distinguished from a dense vector by having a higher percentage of non-zero entries. The embedding space is the name given to this large, high-dimensional region. comparable vectors are allocated to words that have comparable meanings or usage patterns within this area. For example, words like "tea" and "coffee" would probably be close together, but words like "tea" and "sea" would be farther apart because of their different meanings and less common usage, even though they have the same spelling.

Although word embeddings in NLP can be generated in a variety of ways, they are all trained on large amounts of textual data, or what is known as a corpus. This corpus can contain a variety of text kinds; Wikipedia and Google News are two prominent examples that are frequently utilised for pre-trained embedding models. When the pre-trained corpora that are currently available are insufficient, the corpus can be a bespoke embedding layer. During the training process, the algorithm learns to correlate each word with a distinct vector by examining word usage patterns within this data.

3.3.3 *Continuous Bag-of-Words Model (CBOW)*

A common network structure used for embedding words, in natural language processing (NLP) tasks, is the Bag of Words Model (CBOW). The main goal of CBOW is to forecast a target word by examining the context of words within a specified window size (Mikolov et al., 2013).

In the CBOW model, the machine is trained to predict a target word based on the words surrounding it. Unlike language models that anticipate a word in a sequence, CBOW forecasts a word by considering its neighbouring terms. To achieve this, the embeddings of words are averaged to generate a vector representation, which is then utilised for predicting the target word.

Typically, a CBOW architecture consists of three layers, an input layer, a hidden layer and an output layer. The context terms are encoded into the appropriate term embeddings by the input layer. The input to the hidden layer is a single vector representation that is created by averaging these embeddings (Mikolov et al., 2013). After processing this input, the hidden layer sends it to the output layer, where a SoftMax activation function is utilised by the model to predict the target word.

To reduce the loss between the predicted word and the actual target word during training, CBOW modifies the network's weights utilising gradient descent and backpropagation. Maximising the probability of correctly identifying the target expression based on its context is the aim. CBOW is renowned for being more straightforward and effective than other language models, such as Skip-gram (Al Sallab et al., 2015).

All things considered, learning distributed representations of terms in a continuous vector space that picks the semantic ties among expressions based on their contextual usage

is made possible by the Continuous Bag-of-Words Model (CBOW), a potent and popular method.

3.3.4 *Skip-Gram Structure*

The Skip-Gram (SG) mechanism is an extensively implemented neural network architecture for word embeddings in (NLP). (Xiong et al., 2019) describes the SG method as a variation of the Word2Vec strategy, which aims to learn vector representations for phrases relying on their co-occurrence in an enormous dataset of textual data.

A shallow neural network called the SG model forecasts a set context window encircling a given target term. There are three levels in it: an output layer, a concealed layer, and an input layer. The input layer uses a one-hot vector encoding of the target word. Before activation, the hidden layer uses a nonlinear function like tanh to linearly transform the input vector. Every context word in the window is a neuron in the output layer that uses a SoftMax operation to create a probability distribution over the vocabulary that shows how likely it is that any given word will be the context word. To improve the prediction accuracy of context words given the target word, stochastic gradient descent (SGD) modifies the hidden layer's weights during training. Notwithstanding its updating, the output layer's weights are not used for prediction (Preethi Krishna & Sharada, 2020, p. 202). The SG model's hidden layer weights can be used as word embeddings for a variety of natural language processing (NLP) tasks, such as text categorisation, sentiment analysis, and machine translation, once they have been trained. These embeddings reflect word associations in terms of meaning and are derived from phrase co-occurrence within the training corpus (B. Liu, 2022).

Compared to the (CBOW) model, the SG model has advantages, especially when it comes to learning embeddings for uncommon expressions that don't show up regularly in the training set. Moreover, (Xiong et al., 2019) found that the SG model is adept at capturing

verb-object and adjective-noun links, among other syntactic and semantic associations among terms.

3.3.5 *Feature Extraction*

We utilised word embeddings and TF-IDF as our two main, general feature extraction mechanisms. Applying a mechanism that combines TF-IDF weight determines the term importance of each document in a document collection (Ahuja et al., 2019). On the contrary, to using term embeddings as features, we only used TF-IDF weights in our experiments when combined with conventional machine learning techniques.

Applying term embeddings, terms are constituted as dispersed vectors in a high-dimensional space, where terms with diverse meanings are far away and terms with identical meanings are closer together. It is usually possible to learn these embeddings by looking at the contexts in which terms are utilised. Word2Vec is one common mechanism for quickly and effectively learning word embeddings from a textual corpus. For learning embeddings, (Hung & Alfred, 2017) suggest two additional instructional techniques: Skip-gram and (CBOW). In contrast to Skip-gram, which predicts surrounding terms relying on the recent term, the CBOW mechanism predicts the recent term relying on its environment. AraVec2.0 is a pre-trained Arabic term embedding mechanism that we used for our studies. With the help of several datasets, such as web pages, tweets, and Wikipedia Arabic articles, this model provides a variety of topologies (Abdelali et al., n.d.).

In our strategy, we applied the pre-trained approach to apply both traditional and neural learning mechanisms. We created the reviews feature vector by using the average vector of word embeddings in a tweet, which we calculated using standard techniques. In contrast, embedding vectors were used in neural learning models to establish the weights of

the embedding layer, which was subsequently linked to other network layers (Ahuja et al., 2019).

3.3.6 Annotation

Both ML and DL models depend heavily on annotation, which is the procedure of annotating or tagging data to offer insightful information for training models (Ovsiannikov et al., 1999). Annotation in ML and DL is defined as the procedure of manually or automatically assigning labels or annotations to data instances in diverse types, comprising textual, images, voice, and video. This creates a labelled dataset that can be used for training and assessment (Passonneau & Carpenter, 2014). Tagging tasks in the context of picture data can comprise object detection, which entails drawing bounding boxes around particular elements in an image, or image segmentation, in which regions or pixels are annotated to differentiate among various objects or segments. Annotation in image information can also consist of categorisation tasks, in which images are classified into particular groupings or categories. Named entity recognition, which involves locating and labelling items such as names, organisations, or locations inside a text, is one type of annotation activity that can be applied to text data. Analysing texts with emotion categories like positive, negative, or neutral can also be a part of SA. Common uses of annotation in text data include text classification, document categorisation, and sentiment analysis (Zhu et al., 2020).

Human annotators can manually annotate data by carefully going over and labelling it following predetermined standards or criteria. However, this manual method can be costly and time-consuming, particularly for large datasets. Meanwhile, the annotation process can be accelerated via utilising automated or semi-automatic approaches that make use of computer vision algorithms, natural language processing, or crowdsourcing platforms. The effectiveness of machine learning and deep learning models depends heavily on the quality of

annotation, which calls for precise annotation guidelines, metrics for inter-annotator agreement to guarantee consistency among annotators, and continual quality control and feedback systems to maintain high-quality annotations (Qureshi et al., 2022).

Annotation is necessary for the process of labelling data in machine learning and deep learning to create a labelled dataset for training. This is an essential step in the creation of reliable and accurate models, which allow them to recognise patterns and categorise or predict new, invisible data.

The satisfaction scales in this study were designated as negative (N), positive (P) and Neutral (Ne). The annotated comments in the labelled dataset are arranged in accordance with the gathered reviews. Figure 9 below shows an annotated dataset with "P", "N", and "Ne" annotations, respectively. There are two columns in it: "Text" with reviews and "Label" with the satisfaction scale.

Figure 9

Example of the Annotated Dataset

Text	Labbel
انا فرحان بزيارة حضرتك	POS
هذا اسوء عمل شفته في حياتي	NEG
العمل الناجح الدؤب يرفع من شأن الانسان في المجتمع	NON

3.4 Splitting the Dataset

One phase in the ML and DL procedure is to partition a dataset into training, validation, and testing sets. Via this procedure, the system is guaranteed to learn from some data, fine-tune its parameters in another region, and then be tested in another field to examine its efficacy (Ouyang et al., 2022).

1. Training Set: The dataset's segment, which is composed of the training set is applied while instructing the model. The model learns to predict by identifying properties and traits in the data via training. The training set commonly takes about 70–80% of the dataset.

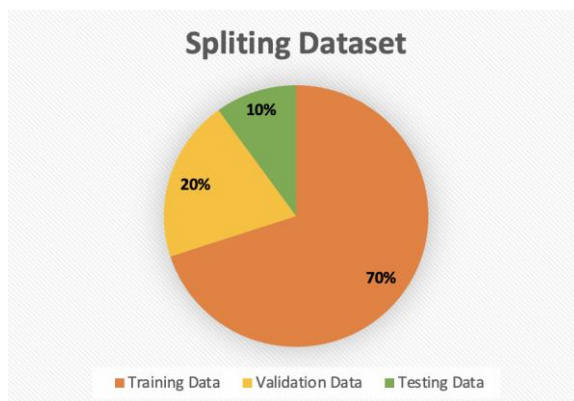
2. Validation Set: The validation set fulfils the role of tuning hyperparameters. Monitoring how good the mechanism performs during training. It serves as a guard against overfitting by supplying a dataset that was not part of the model's training operation. The validation set assists in selecting the performing model based on its efficacy, with data. Usually, 10- 15% of the data is reserved for this purpose.

3. Testing Set: The trained technique's final functioning is examined using the testing set. It offers a fair inspection of the model's efficiency using hypothetical data. Usually, the testing set receives the remaining 10–15% of the data.

In this dissertation, the Arabic corpora either text or speech collected for analysis were divided into training, validating, and testing datasets, respectively, using a 70:20:10 ratio. Figure 10 shows a flow chart for the training, validation and testing data for the text and speech reviews.

Figure 10

The Dataset Split



3.5 Deep Learning Models

Assessing people's opinions about a good or service is important for determining how satisfied customers are with the service they receive. Online shopping is currently a forum for opinion expression, thus, information from these platforms was gathered and examined to support choices meant to increase customer happiness and loyalty rates. This work provides deep learning (DL) models intended to evaluate positive, negative and neutral feelings expressed in people's opinions regarding any topic reviews in the Arabic regions in order to examine a significant volume of these opinions. The classification results were then compared using a variety of deep learning models, such as Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (BI-LSTM), Generative Pre-trained Transformers (GPT) and Gated Recurrent Unit (GRU).

3.5.1 *Convolution Neural Network (CNN)*

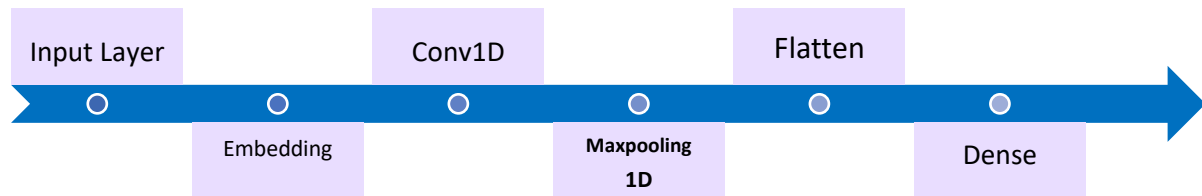
Convolutional neural networks (CNNS) are a robust deep learning method for missions including image identification and computer vision. CNNS are influenced by the way the brain works with visual data. They were created especially to find patterns and characteristics in photographs on their own (H. Kim & Jeong, 2019). CNNS are a calculation mechanism that activates the neural networks located in the brain. Whilst specific systems just make use of the layers of input and output, other people can also use the three fundamental levels—the input, output, and hidden layer. In the last few years, several studies have investigated NLP in conjunction with (SA), with encouraging results.

(CNN) are famous for their capability to learn levels of features from input data. Simple characteristics like edges and patterns are grasped by the layers, while more intricate aspects such as objects and shapes are understood by the layers. Due to this structure, CNNs excel in tasks related to image and video recognition. However, it's important to mention that

training a network (CNN) requires a substantial amount of labelled data and can be both time-consuming and resource-intensive. Additionally, interpreting how CNNs arrive at their decisions can be challenging, given their "black box" reputation (Albawi et al., 2017). As a result, there have been efforts to develop methods for evaluating and understanding these processes.

CNNs are an arithmetic formulation that activates the brain's neural networks. While some systems just make use of the input and output layers, others can also use the three fundamental levels—the input, output, and hidden layers. In the last few years, several studies have investigated NLP in conjunction with sentiment analysis, with encouraging results. The CNN and LSTM algorithms outperformed the single model. Maximal Entropy, Support Vector Machines, and Naive Bayes. To represent the grammar as a collection of words, CNN used a single convolutional layer (Badshah et al., 2017).

CNNs can learn levels of characteristics from input data. They start by grasping elements like edges and textures at the levels and then progress to more complex features, such as objects and shapes at higher layers. This hierarchical structure makes CNNs effective in tasks involving image and video recognition. However, training a network needs a significant amount of labelled data and can be both time-consuming and resource-intensive. Additionally, understanding how CNNs make decisions can be challenging due to their reputation as "boxes." As a result, efforts have been made to develop methods for evaluating and interpreting their predictions effectively. Figure 11 depicts the CNN architecture.

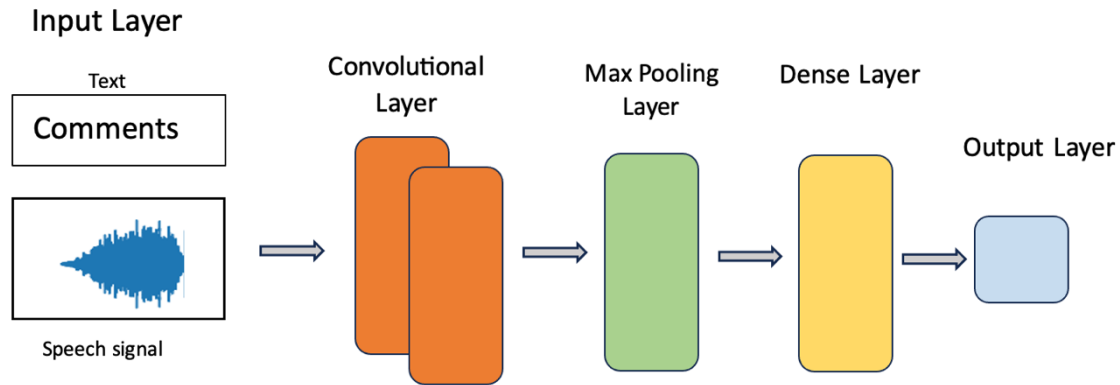
Figure 11*CNN Framework*

It should be observed, however, that (CNN) training requires a huge size of annotated dataset and can be resource-intensive and time-consuming. Furthermore, understanding the logic underlying CNN predictions may be difficult due to their classification as "black boxes" (Albawi et al., 2017). As such, efforts have been undertaken to offer techniques for the analysis and visualisation of the patterns that Convolutional Neural Networks (CNNs) learn. In Figure 12, the CNN Architecture is shown. Several hidden layers are arranged between the input and output layers of a (CNN).

Convolution: Convolution applies a series of filters to process the input images. Every filter recognises a unique characteristic and turns it on or off in response to it.

Rectified Linear Units (ReLU): provide for faster and more efficient training by maintaining positive values and setting negative values to zero. Simply said, "activation" refers to the transfer of only the active attributes to the next layer.

Pooling: Pooling uses nonlinear down-sampling to simplify the output and decrease the number of parameters the network has to learn.

Figure 12*CNN Architecture*

3.5.2 The Long Short-Term Memory (LSTM)

One limitation of recurrent neural networks (RNN) is their restricted capability to deal with contextual information. Therefore, learning contextual dependencies with a long-term nature is challenging when using RNN. This difficulty arises from the recurrent connection, which causes the effect of the inputs to either decay or balloon exponentially (Le et al., 2019). Due to the update operation, this challenge, known as vanishing gradient, causes incorrect gradients. LSTM is designed as an improved version of RNN to address the vanishing gradient issue with RNN. When used to handle SA challenges, LSTM has demonstrated good results (Wang et al., 2018). Memory blocks, which are self-linked subnetworks comprising several internal cells, are a component of the hidden layer in LSTM.

Memory blocks are applied by the LSTM neural network to replace the hidden units found in classical RNNs. Each memory block has triple multiplicative gates (forget, output, and input) (Zazo et al., 2016) in addition to one or more memory cells that are randomly linked. The input gate chooses which part of the input stream will be applied to update the cell state. The output gate selects the portion of the newly formed cell state that will be the

output. The forget gate selects which portion of the previous cell state will be erased. The LSTM's memory block is shown in Figure 4.11(Luo, 2020).

The time step is indicated by the symbol t in the memory block. The output, input, and forget gates are represented by the symbols o_t , i_t , and f_t , respectively. The input and the output at this phase are indicated by the symbols h_t and x_t . h_{t-1} represents the output at the previous time step, while σ represents the sigmoid activation function. The symbol $\otimes$ is used to symbolise the Hadamard product. Below are the primary formulas that depict the LSTM scheme.

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)$$

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i)$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o)$$

$$\tilde{c}_t = \tanh(W_x x + W_h h + b)$$

$$c_t = f \otimes c + i \otimes \tilde{c}$$

$$h_t = o_t \otimes \tanh(c_t) \tag{1}$$

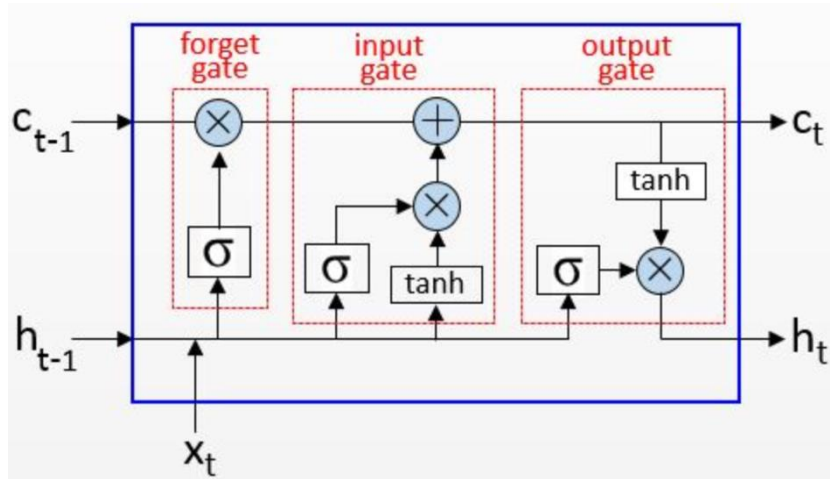
Figure 13 shows the LSTM memory block (*Recurrent Neural Network (RNN) and LSTM | Data Platform and Data Science, n.d.*).

The weight matrices are represented by the letter W for the three gates, while the biases are represented by the symbol b . The input, forget, and output gates are represented by the subscripts of the memory cells, o , i , and f , respectively. The memory cell is indicated by the subscript c . The information updating process by the gate structure might be divided into three levels after the computation of the three gate values. The first level begins with

multiplying the two numbers associated with c_{t-1} (old cell state) and f_t (forget gate) to determine the portion of the previous cell state that will be discarded.

Figure 13

LSTM Memory Block



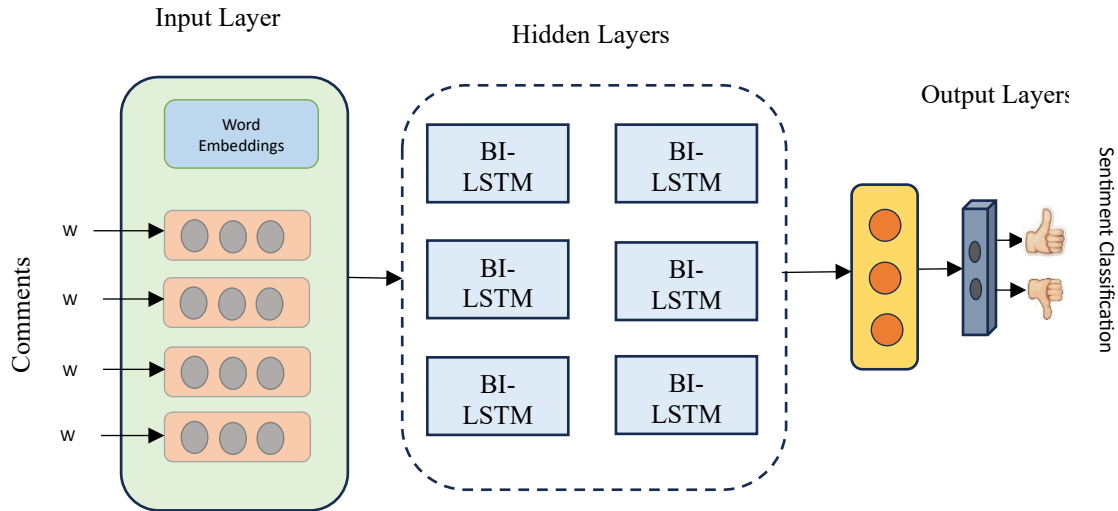
The second level begins with multiplying the two values associated with $\tilde{c}$ (new candidate memory t cell) to update the cell state information. To obtain h_t (output value), the third level starts by multiplying the two values pertaining to c_t (updated cell state) and o_t (output gate) by the tanh function. The values of c_t and h_t will transition to the next memory block at time step $t + 1$.

3.5.3 Bidirectional Long Short-Term Memory BI-LSTM

Bidirectional LSTM networks can handle sequential data in a comprehensive manner with contextual awareness from both sides since they can take into account both past and future data (Zhang et al., 2015). 'Encoding' is the first step in converting the input sequence into numerical representations. The encoded input sequence is processed step by step by a forward LSTM layer, which records the historical background. On the other hand, a backwards LSTM layer, like a forward LSTM, reverses the encoded input sequence and

modifies its internal memory state according to input and memory states at each time step. Bidirectional LSTM networks, by merging forward and backwards LSTM layers, obtain a comprehensive understanding of the input sequence at any given moment. Routing the output layer through completely connected layers produces the final output (Efanov et al., 2022). Bidirectional LSTM networks are frequently chosen for sequence modelling because of their improved performance and capacity to capture bidirectional dependencies, despite their higher training and inference costs (Nguyen et al., 2016).

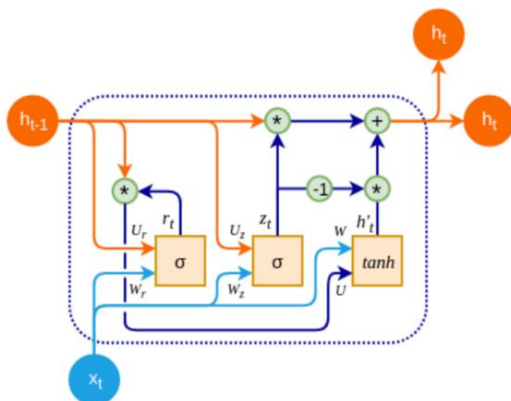
As shown in Figure 14, the Bi-LSTM model consists of multiple crucial elements for text analysis. Every phrase or token is transformed into a dense vector in continuous space by an embedding layer. Each word in the input text is represented as a 64-dimensional semantic vector in this model because the embedding layer used a vocabulary of $\text{Len}(\text{tokeniser.index_word}) + 1$ and 64 embedding dimensions. A 128-unit LSTM bidirectional layer that captures long-term dependencies and preserves memory state comes next, and it is designed to process sequential input optimally. The bidirectional layers examine the input sequences, and `Return sequences = True` guarantees that the layer preserves the natural order of the data by returning the hidden state output for each time step in the series. Next, a second 32-unit LSTM bidirectional layer is applied, which extracts contextual data from the input sequence's forward and backward motions. Finally, using the contextual information from the input sequence, the fully connected dense layer predicts two classes: satisfactory and unsatisfactory. The collected values are passed through the thick layer to determine the probability for each class, which ranges from 0 to 1 (Zhang et al., 2015).

Figure 14*BI-LSTM Architecture*

3.5.4 Gated Recurrent Unit (GRU)

Recurrent neural network (RNN) structures such as the (GRU) are significantly employed in (DL) for serial data analysis. A version of the classic RNN framework named GRU was created to reduce its drawbacks, particularly the problem of gradients decreasing.

Figure 15 represents the cell of the GRU (Cho et al., 2014).

Figure 15*The Cell of GRU*

The GRU model's salient characteristics and elements are as follows:

1. **Gating Function:** GRU has a gating function to help control the data stream inside the system. The reset gate and the update gate are its constituent gates (Bhatta et al., 2020).
2. **Reset Gate:** This mechanism specifies the amount of historical data that must be reset or forgotten. it produces a number between 0 and 1 for every component in the hidden layer after receiving data from the step that preceded it and the present input (Fargalla et al., 2024).
3. **Update Gate:** This mechanism regulates the range of historical data that is kept and the amount of fresh input that is provided. In addition, it receives input from the present time step and the preceding one, and for each unit in the hidden state, it outputs a value between 0 and 1.
4. **Hidden State:** The GRU's hidden state stores the knowledge gleaned from preceding time steps and serves as an illustration of the network's memory. Every time step, it is modified in accordance with the input, the gating function, and the prior hidden condition (Irie et al., 2016).
5. **Training:** Stochastic gradient descent (SGD) or Adam are examples of gradient descent optimisation techniques that are used to train GRU and other neural network systems. As the model undergoes training, it learns how to reduce the degradation function—a measurement of the discrepancy between the expected and actual outputs—by adjusting the parameters, or weights and biases (Girirajan & Pandian, 2022).
6. **Applications:** GRU has been effectively used for a number of sequential data processing applications, such as speech recognition, time series prediction, video analysis, and natural language processing (NLP). Tasks where long-term

dependencies are crucial can benefit greatly from it, but modelling them with conventional RNNs can be computationally costly (Bhatta et al., 2020).

7. Benefits: Compared to conventional RNNs, GRU has a number of advantages, including quicker training durations, superior handling of vanishing gradients, and enhanced performance on applications requiring lengthy data sequences (Fargalla et al., 2024).

3.5.5 *Generative Pre-trained Transformer (GPT)*

OpenAI constructed a Generative Pre-trained Transformer (GPT), an innovative computational language structure. It is a member of the transformer-based framework group, which has transformed jobs related to (NLP) because of its capacity to extract contextual and long-range relationships from linguistic input (Vaswani et al., 2017). Here are some of the major characteristics and aspects of GPT:

Transformer Architecture: The transformer architecture described in the article (Vaswani et al., 2017) serves as the foundation for GPT. Transformers efficiently capture context by encoding and decoding information based on the relative relevance of phrases in a sequence via self-attention mechanisms.

Pre-training: Using unsupervised learning objectives like language modelling or subsequent word Recognition, GPT algorithms are pre-trained on sizable text data corpora. In pre-training, the framework gains a deep understanding of language patterns and semantics by learning to anticipate the next word in a series given the previous context (X. Liu et al., 2023).

Generative: founded on an instruction or seed words, GPT is a generative technique that may create terminology that is both consistent and context suitable. Through the use of

the acquired language representations, GPT can create grammatically sound and semantically pertinent writing that resembles humans (Vaswani et al., 2017).

Fine-tuning: GPT mechanisms can be enhanced by utilising labelled data on particular downstream tasks following pre-training on a sizable dataset. In order to adjust the mechanism parameters for the goal task, such as text classification, question answering, or text generation, fine-tuning entails updating the model. With some fine-tuning, GPT can perform at the cutting edge on a variety of NLP tasks (X. Liu et al., 2023).

Scalability: GPT can be implemented in a variety of algorithms with billions of factors, spanning from tiny to extra-large. Larger models need more computing power for training and inference, but they also perform better on downstream tasks and are able to capture more complicated patterns in language.

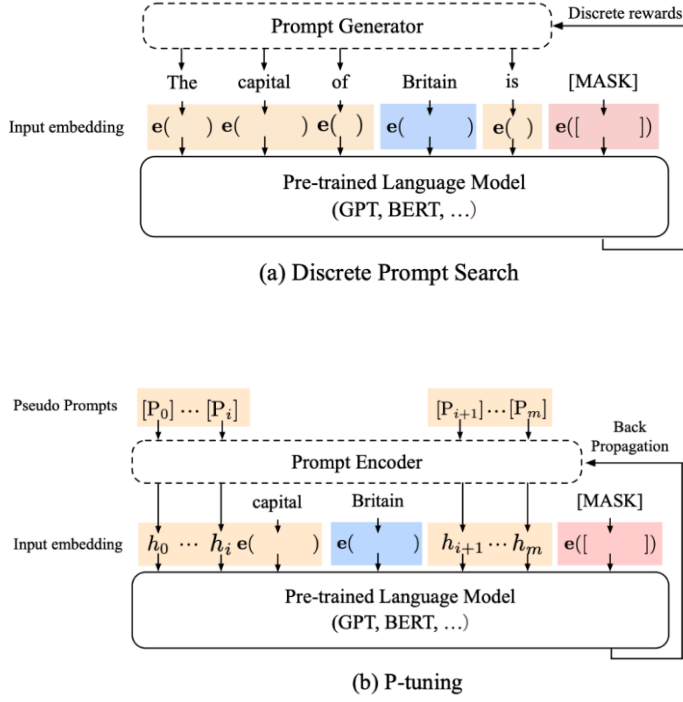
Applications: GPT is extensively utilised for a variety of (NLP) activities, such as SA, summarised texts, language translation, dialogue generation, and code generation. It is a flexible tool suitable for a variety of uses due to its capacity to produce content that is both cohesive and contextually relevant.

All things considered, GPT has made a substantial contribution to the field of NLP by proving that transformer-based architectures are efficient at capturing intricate language patterns and producing text of excellent quality. GPT and its variations keep pushing the limits of what is feasible in linguistic production and interpretation tasks through ongoing research and development. Figure 16 shows an example of a prompt search (X. Liu et al., 2023). From the figure, an illustration of a quick search for "[MASK] is the capital of Britain" The orange zone refers to the prompt given the context (blue zone, "Britain") and target (red zone, "[MASK]"). The prompt encoder and continuous prompt embeddings in (b)

can be optimised in a differentiable manner than the prompt generator in (a), which only receives discrete rewards.

Figure 16

Example of a Prompt Search (Liu et al., 2023).



3.6 Evaluation Metrics

Metrics for assessment are significant to evaluating the efficacy as well as the success of data analysis and ML techniques. They act as impartial yardsticks for assessing how well a model performs its intended function, be it classification, regression, or any additional analytical or predictive activity. These evaluations are important, especially when it comes to the field of sentiment analysis applied to Arabic sentiment analysis, such as comments evaluations. In order to assess the performance of the system in analyzing and classifying the attitudes stated in the reviews, they provide unbiased benchmarks to calculate accuracy, recall, precision, and f1 score (Sokolova et al., 2006).

Accuracy: Accuracy evaluates how well the approach predicts things in general by dividing the total amount of comments that are properly categorized by the number of comments. Because accuracy provides a broad assessment of the algorithm's efficiency, it might be misleading when applied to an unbalanced corpus. Accuracy, which measures the percentage of reviews that were accurately predicted out of all the cases in the dataset, is widely acknowledged as the most basic and common parameter for classification algorithms. According to (Sokolova et al., 2006), accuracy essentially indicates the model's capacity to classify examples into the appropriate groups. The following is the computation made with Eq. (2):

Accuracy = (true positives + true negatives) / (true positives + false positives + true negatives + false negatives)

$$Accuracy = \frac{TN+TP}{TN+FP+TP+FN} \quad (2)$$

Precision is defined as the ratio of correctly predicted positive cases to all positive cases that were forecasted. It highlights the model's ability to reduce false positives, which is crucial when there are serious repercussions from false positives. Precision measures the percentage of real positives among all cases that the model predicts to be positive. In essence, it assesses how well the model predicts good outcomes. Eq. (3) is used to compute precision, as shown below:

Precision = true positives / (true positives + false positives)

$$Precision (P) = \frac{TP}{TP+FP}$$

(3)

Recall: often identified as True Positive Rate or Sensitivity, this metric measures the proportion of true positive forecasts to all real positive cases as in Eq. (4).

Recall = true positives / (true positives + false negatives)

$$Recall (R) = \frac{TP}{TP+FN}$$

(4)

The F1-score, considered to be the harmonic mean of recall and precision, the F1-score provides a fair evaluation of both measurements. It is beneficial in circumstances when recall and precision play a role, as it incorporates the benefits of both measures. The F1 score computation is shown in Eq. (5) as follows:

F1 score = 2 * (precision * recall) / (precision + recall)

$$F1 - score = \frac{2*P*R}{P+R} \quad (5)$$

A confusion matrix is an essential instrument utilised in statistics and ML classification for assessing how well a classification method is performing. When dealing with binary or multiclass classification problems, it is very helpful since it offers a thorough picture of the system's performance. By dividing the expected and actual class labels into separate groups, the confusion matrix provides a concise overview of the classification procedure.

Important Elements:

1. True Positives (TP): Cases in which the positive class is correctly predicted by the model.

2. True Negatives (TN): Cases in which the model correctly forecasts the positive class's absence.

3. Type I errors, or False Positives (FP): When the model forecasts the positive class inaccurately.

4. False Negatives (FN)-Type II Error: Situations in which the model forecasts the negative class incorrectly.

Confusion Matrix Structure: An $N \times N$ matrix, where N is the total number of classes, and a 2×2 matrix for binary classification are the two typical configurations for confusion matrices.

Classification in binary form: For binary classification, Table 6 shows how the confusion matrix is set up.

Table 8

Confusion Matrix Binary Classification

	Predicted Positive	Predicted Negative
Actual Positive	True Positives	False Negatives
Actual Negative	False Positives	True Negatives

3.7 Summary

A brief description of the methodology employed in this dissertation has been supplied. The processes involved in collecting and processing data, such as feature extraction, data partitioning, and annotation, are thoroughly explained in this section. This chapter also looks at a number of deep learning architectures, such as GPT techniques, Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (BI-LSTM), and Convolutional

Neural Networks (CNNs). Lastly, the section offers a thorough justification of the evaluation metrics applied in the study.

Chapter 4: Implementation and Experiment

4.1 Setup and Configuration

The execution of the suggested mechanism requires the incorporation of various tools and algorithms, and adopting a suitable framework for the sound and text Arabic sentiment analysis system advancement is essential. The following libraries and implementation utilities were utilised: below is an overview of each of the libraries that are implemented in this dissertation:

1. *****Pandas*****:

- Pandas is a potent information processing and examination library for Python.
- It offers data structures like DataFrame and Series, ideal for working with structured data.
- Pandas offers functionalities for perusal and scripting information from a variety of file styles, data cleaning, reshaping, grouping, and accumulating.

2. *****NumPy*****:

- NumPy is an essential package for mathematical computing in Python.
- It supplies support for multi-dimensional arrays and matrices, along with a collection of mathematical techniques to operate on these arrays efficiently.
- NumPy is commonly utilised in scientific computing, machine learning, and data analysis due to its speed and efficiency.

3. *****Librosa*****:

- Librosa is a Python module for audio and music signal treatment.

- It allows users to load audio files, and extract various audio features such as Mel-Frequency Cepstral Coefficients (MFCCs), chroma features, and rhythm properties.

- Librosa offers tools for time-series analysis, representation, and manipulation of audio data.

4. *****scikit-learn (sklearn)*****:

- Scikit-learn is a common (ML) library for Python.
- It provides simple and efficient tools for data mining and data evaluation, consisting of supervised and unsupervised learning techniques.
- Scikit-learn offers functions for classification, regression, clustering, dimensionality reduction, model selection, and preprocessing.
- It is known for its easy-to-use interface and extensive documentation.

5. *****TensorFlow*****:

- TensorFlow is an open-source (ML) framework developed by Google Brain.
- It offers tools for building and training (DL) models, including neural networks and other computational graphs.
- TensorFlow supports both CPU and GPU computing and offers high-level APIs like Keras for building and training models with ease.
- It is widely used in various fields such as computer vision, NLP, and reinforcement learning.

6. ***Seaborn***:

- Seaborn is a Python visualisation library based on matplotlib.
- It supplies a high-level interface for creating attractive and informative statistical graphics.
- Seaborn simplifies the process of creating sophisticated visuals such as heatmaps, violin plots, pair plots, and time series plots.
- It offers built-in themes and colour palettes to enhance the aesthetics of the plots.

7. ***Matplotlib***:

- Matplotlib is a comprehensive library for creating static, animated, and interactive visuals in Python.
- It provides a MATLAB-like interface for plotting 2D and 3D graphics.
- Matplotlib can be used to create a wide range of plots, including line plots, scatter plots, bar plots, histograms, and contour plots.
- It is highly customizable and offers fine-grained control over every aspect of the plot's appearance.

These libraries are essential tools for various information examination, (ML), and visual tasks in Python, and mastering them can significantly develop the capabilities of a data scientist or (ML) practitioner.

8. Google Colab: formerly referred to as Google Colaboratory, is an online platform where individuals may run Python code and access GPU and TPU resources for free. It lets users develop and run Python code in a web-based environment and is built on top of Jupyter

Notebooks. Because it provides a straightforward way to execute code, visualise data, and work with others without requiring any setup or installation, Colab is frequently used for an assortment of data mining and (ML) projects.

4.2 Experimentation Construction

4.2.1 Collecting and Setting Up the Dataset

The precise data collection and formatting procedures used in the study guaranteed the integrity and reliability of the Arabic voice emotion recognition dataset. A summary of the methods utilised to collect and process the data is given in this section.

- **Data Collection**

To acquire Arabic audio and text comments, data was gathered from several web pages, including pages on YouTube and Instagram. To get a variety of reviews from numerous sites, the process of acquiring reviews was thorough. This method covered a broad spectrum and the related texts of sentiment analysis and emotional categories, ensuring the dataset's inclusivity. Kindly see part 3.1 for further information.

- **The Process of Preparing the Data**

Data preparation is a core level in pre-treating data for examination and (ML) programs. This technique requires turning unprocessed, raw data into a refined, standardised format that is ideal for effective algorithmic application. The quality and precision of analytical results are significantly enhanced by this first stage. As outlined in sections 3.2 to 3.6, several preprocessing procedures were carried out on the unprocessed data to get it ready for analysis and categorisation.

4.2.2 *Applied Deep Learning Models*

This dissertation provides (DL) models intended to evaluate positive, negative and neutral feelings expressed in people's opinions regarding product reviews and customer service reviews in Arabic regions to examine a significant volume of these opinions. The classification results were then compared using a variety of deep learning models, such as Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (BI-LSTM), Generative Pre-trained Transformers (GPT) and Gated Recurrent Unit (GRU).

- ***Convolution Neural Network***

Convolutional Neural Networks (CNNs) are a classifier of deep neural network that have accomplished remarkable outcomes in several realms of computers, such as computer visions, NLP, speech recognition, and medical image analysis.

Table 7 below shows the hyperparameters of the CNN Model.

Table 9

The Hyperparameters of the CNN Classifier

Layer	Hyper-parameter	Values for the Text	Values for the speech
Convolutional Layer	Number of filters	128	128
	Kernel size	5	3
	Activation function	ReLU	ReLU
	Padding	'same'	'same'
Max Pooling 1D Layer	This layer performs global max pooling operation over the entire sequence		
	pool_size	2	2
Dense Layer	Number of units	128	128
(Hidden Layer)	Activation function	ReLU	ReLU
Dropout Layer	The Dropout rate	0.5	0.2
Dense Layer	Number of units	128	128
(Hidden Layer)	Utilized Activation function	ReLU	ReLU

Dense Layer (Output Layer)	Number of units	3	3
	Utilized Activation function	Softmax	Softmax
	The epochs number	10	100
	The batch_size	32	512

This comparison table outlines the specific hyperparameters and values used for each layer in the (CNN) architecture for processing text and speech data. It highlights the differences in settings between the two datasets, such as the number of filters, kernel size, dropout rate, and batch size, which may be tailored based on the features and requirements of each data type. Additionally, it shows the consistency in activation functions and layer configurations across both text and speech processing tasks.

Following is an illustration of the CNN model including the hyperparameters and the layers in the model:

In the provided code, the CNN model consists of the following hyperparameters and layers:

1. ****Embedding Layer****:

- The embedding layer converts integer-encoded vocabulary indices into dense vectors of fixed size.
- Hyperparameters:
 - ``input_dim``: Size of the vocabulary, calculated as the length of the tokenizer's word index plus one.
 - ``output_dim``: Dimensionality of the dense embedding vectors. Here, it is set to 200.
 - `- `input_length``: Length of input sequences after padding to a fixed length (``max_len``).

2. **Convolutional Layer**:

- The convolutional layer applies convolutional filters to the input text sequences.
- Hyperparameters:
 - ``filters``: Number of filters (or kernels) to convolve over the input sequence. Set to 128 in both systems.
 - ``kernel_size``: Size of the convolutional filters. For the text, it is set to 5, while the speech recognition is set to 3.
 - ``activation``: Activation function applied after the convolution operation, set to 'relu' for both systems.
 - ``padding``: Padding mode for the convolution operation, set to 'same' to ensure that the output has the same length as the input in both systems.

3. **GlobalMaxPooling1D Layer**:

- The global max pooling layer performs down-sampling by taking the maximum value across all time steps (along the sequence dimension).
- This reduces the dimensionality of the feature maps while preserving important information.

4. **Dense Layers**:

- Dense layers are fully connected layers that perform classification based on the features extracted by the convolutional layers.
- Two dense layers are added with 128 units each and ReLU activation functions in both systems.

- A dropout layer is inserted between the dense layers with a dropout rate of 0.5 to prevent overfitting for the text system, while the speech recognition, the dropout rate is 0.2.
- The final dense layer has 3 units (equal to the number of output classes) and uses a softmax activation function for multi-class classification.

5. **Optimizer**:

- The model is compiled using the Adam optimiser, a popular choice for gradient-based optimisation algorithms.

6. **Loss Function**:

- The loss function used is sparse categorical cross-entropy, which is suitable for multi-class classification tasks where the labels are provided as integers.

7. **Metrics**:

- The accuracy metric is used to monitor the performance of the model during training.

These hyperparameters and layers define the architecture of the CNN model for text classification. The model is then trained on the training data and evaluated on the test data using classification metrics like precision, recall, and F1-score.

- ***Long Short-Term Memory***

Table 8 below shows the hyperparameters of the LSTM Model for both text and speech sentiment analysis. This comparison highlights the differences in model architecture and hyperparameter settings between text and speech processing tasks, reflecting the unique characteristics and requirements of each type of data.

Table 10*The LSTM Hyperparameters*

Layer	Hyper-parameter	Value for the Text	Value for the speech
LSTM Layer	The Number of utilised units	40	40
	Return sequences:	False (for the first BI-LSTM layer)	False (for the first BI-LSTM layer)
Dense Layer (Hidden Layer)	The Number of used units	128	64
	The type of Activation function	ReLU	ReLU
Dropout Layer	The number of the Dropout rate	0.5	0.2
Dense Layer (Hidden Layer)	The number of used units	128	32
	The type of Activation function utilised	ReLU	ReLU
Dense Layer (Output Layer)	The number of applied units	3	3
	The type of Activation function applied	Softmax	Softmax
	The number of epochs	10	100
	The size of an applied batch	32	512

Below is an explanation of the LSTM, which implements a Long Short-Term Memory (LSTM) model for a text classification task alongside the speech recognition for the same reviews, along with a description of the layers used:

1. ****Embedding Layer****:

- This layer converts integer-encoded vocabulary indices into dense vectors of fixed size (embedding vectors).
- Hyperparameters:
- ``input_dim``: Size of the vocabulary, calculated as the length of the tokenizer's word index plus one.

- ``output_dim``: Dimensionality of the dense embedding vectors. Here, it is set to 200.
- ``input_length``: Length of input sequences after padding to a fixed length (``max_len``).

2. ****LSTM Layer****:

- The long Short-Term Memory (LSTM) layer is a type of recurrent neural network (RNN) layer that is capable of learning long-term dependencies.
- Hyperparameters:
 - ``units``: Number of units (or neurons) in the LSTM layer. Here, it is set to 128 for the text, and 64 for the speech recognition.
 - ``dropout``: Dropout rate for input units, controlling the dropout regularisation. It is set to 0.5 for the text and 0.2 for the speech recognition.
 - ``recurrent_dropout``: Dropout rate for recurrent units, controlling the dropout regularisation. It is also set to 0.5 and 0.2 for speech recognition.

3. ****Dense Output Layer****:

- This is a fully connected dense layer with softmax activation, which produces the final output probabilities for each class in both systems.
- Hyperparameters:
 - ``units``: Number of units (or neurons) in the dense layer. Set to the number of output classes, which is 3 (assuming a multi-class classification task) in both systems.
 - ``activation``: Activation function applied to the output layer, which is softmax for multi-class classification in both systems.

These layers collectively form the LSTM model, which is then compiled, trained, and evaluated for the text classification task and speech recognition system.

- ***Bidirectional Long Short-Term Memory***

Table 9 below shows the hyperparameters of the BI-LSTM Model for both text and speech sentiment analysis. This comparison highlights the differences in model architecture and hyperparameter settings between text and speech processing tasks, reflecting the unique characteristics and requirements of each type of data.

Table 11

Hyperparameters for the BI-LSTM Model

Layer	Hyper-parameter	Value for the Text	Value for the speech
BI-LSTM Layer	The Number of utilised units	40	40
	Return sequences:	False (for the first BI-LSTM layer)	False (for the first BI-LSTM layer)
Dense Layer (Hidden Layer)	The Number of used units	64	128
	The type of Activation function	ReLU	ReLU
Dropout Layer	The number of the Dropout rate	0.5	0.2
Dense Layer (Hidden Layer)	The number of used units	128	32
	The type of Activation function utilised	ReLU	ReLU
Dense Layer (Output Layer)	The number of applied units	3	3
	The type of Activation function applied	Softmax	Softmax
	The number of epochs	10	100
	The size of applied batch	32	512

The previous table compares the characteristics of a BI-LSTM model, particularly focusing on the hyperparameters and settings for text and speech processing tasks:

Characteristics of the Table

1. BI-LSTM Layer:

- Utilises 40 units for both text and speech processing.
- Return sequences are set to False for the first BI-LSTM layer in both cases.

2. Dense Layer (Hidden):

- For text processing, 64 units are used with ReLU activation function.
- For speech processing, 128 units are used with ReLU activation function.

3. Dropout Layer:

- The dropout rate is set to 0.5 for text processing and 0.2 for speech processing.

4. Dense Layer (Hidden):

- For text processing, 128 units are used with ReLU activation function.
- For speech processing, 32 units are used with ReLU activation function.

5. Dense Layer (Output):

- Applies 3 units with softmax activation function for both text and speech processing.

Other:

The number of epochs is set to 10 for text processing and 100 for speech processing.

The size of the applied batch is 32 for text processing and 512 for speech processing.

This table provides a clear comparison of the key characteristics and hyperparameters of the BI-LSTM model for text and speech processing tasks, highlighting the differences in settings tailored to each type of data.

- ***Gated Recurrent Unit (GRU)***

Table 10 below shows the hyperparameters of the (GRU) Model for both text and speech sentiment analysis. This comparison highlights the differences in model architecture and hyperparameter settings between text and speech processing tasks, reflecting the unique characteristics and requirements of each type of data.

Table 12

Hyperparameters for the GRU Model

Layer	Hyper-parameter	Value for the Text	Value for the speech
Dense Layer	The Number of used units	128	128
(Hidden Layer)	The type of Activation function	ReLU	ReLU
Dropout Layer	The number of the Dropout rate	0.5	0.2
Dense Layer	The number of used units	128	32
(Hidden Layer)	The type of Activation function utilised	ReLU	ReLU
Dense Layer	The number of applied units	3	3
(Output Layer)	The type of Activation function applied	Softmax	Softmax
	The number of epochs	10	100
	The size of applied batch	32	512

The provided table outlines the characteristics of a neural network model, specifically focusing on the hyperparameters and settings for text and speech processing tasks.

Characteristics of the Table

1. Dense Layer (Hidden):

- Utilises 128 units for both text and speech processing.
- ReLU activation function is applied to both layers.

2. Dropout Layer:

- The dropout rate is set to 0.5 for text processing and 0.2 for speech processing.
3. Dense Layer (Hidden):
 - For text processing, 128 units are used with ReLU activation function.
 - For speech processing, 32 units are used with ReLU activation function.
 4. Dense Layer (Output):
 - Applies 3 units with softmax activation function for both text and speech processing.
 5. Other: The number of epochs is set to 10 for text processing and 100 for speech processing. The size of the applied batch is 32 for text processing and 512 for speech processing.

This table provides a concise overview of the key hyperparameters and settings for a neural network model designed for text and speech processing tasks. It highlights the differences in model architecture and training configurations tailored to each type of data.

- ***Generative Pre-trained Transformer (GPT)***

During the implementation of GPT (Generative Pre-trained Transformer), numerous errors emerged throughout the training process, resulting in challenges in accomplishing both text and speech tasks.

4.3 Findings and Discussions

The outcomes from comparing Arabic speech sentiment analysis with Arabic text sentiment analysis implementing CNN, LSTM, BI-LSTM, GRU and GPT, offer precious insights into how well these algorithms function and their efficiency within these particular realms.

Table 13 and Figure 17 are for the showing the performance of the CNN model in classifying both Arabic speech and Text sentiment.

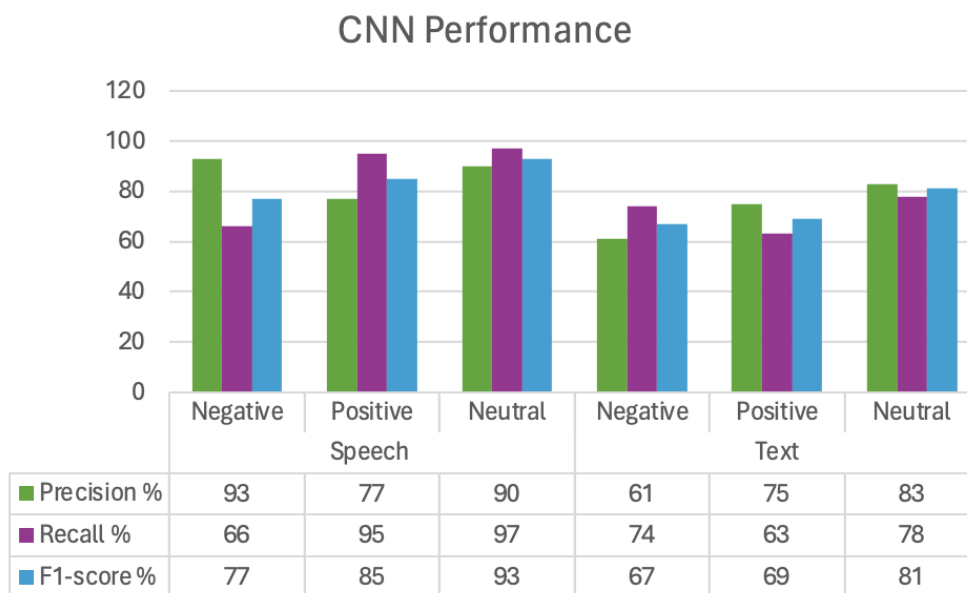
Table 13

The Performance of CNN on Speech and Text Datasets

Dataset	Classes	Precision %	Recall %	F1-score %
Speech	Negative	93	66	77
	Positive	77	95	85
	Neutral	90	97	93
Text	Negative	61	74	67
	Positive	75	63	69
	Neutral	83	78	81

Figure 17

CNN Performance



The previous table and figure present the performance metrics for sentiment classification in two datasets: Speech and Text. Across both datasets, sentiments are categorised into three classes: Negative, Positive, and Neutral.

In the Speech dataset, the Negative class achieved a Precision of 93%, Recall of 66%, and F1-score of 77%, meanwhile, the Positive class exhibited a Precision of 77%, Recall of 95%, and F1-score of 85%, while the Neutral class achieved a Precision of 90%, Recall of 97%, and F1-score of 93%.

In contrast, the Text dataset displayed different results, where the Negative class had a Precision of 61%, Recall of 74%, and F1-score of 67%, while the Positive class showed a Precision of 75%, Recall of 63%, and F1-score of 69%, and the Neutral class attained a Precision of 83%, Recall of 78%, and F1-score of 81%.

These results indicate differing levels of performance between the Speech and Text datasets. While the Speech dataset generally achieved higher performance metrics across all sentiment classes, the Text dataset demonstrated a more mixed performance, with lower Precision and Recall values for the Negative and Positive classes compared to the Speech dataset. These variations highlight the importance of considering dataset characteristics and preprocessing techniques in sentiment analysis tasks.

The results in the table illustrate notable differences in sentiment classification performance between the Speech and Text datasets. In the Speech dataset, sentiments are generally classified with higher Precision, Recall, and F1-scores across all classes compared to the Text dataset. Particularly, the Speech dataset exhibits stronger performance in identifying Negative and Positive sentiments, with Precision values exceeding 75% for all classes. Conversely, the Text dataset demonstrates more varied performance, with lower Precision and Recall values observed for the Negative and Positive classes. However, the Text dataset achieves higher Precision and Recall for the Neutral class compared to the Speech dataset. These disparities suggest that while the Speech dataset may excel in certain sentiment classification tasks, the Text dataset showcases a more balanced performance

across sentiment classes. Such insights emphasise the importance of dataset selection and preprocessing methods in sentiment analysis endeavours.

Figures 18 and 19 below display the confusion matrix for both Arabic speech and text sentiment analysis, respectively, using a CNN model.

Figure 18

Confusion matrix for speech SA using CNN

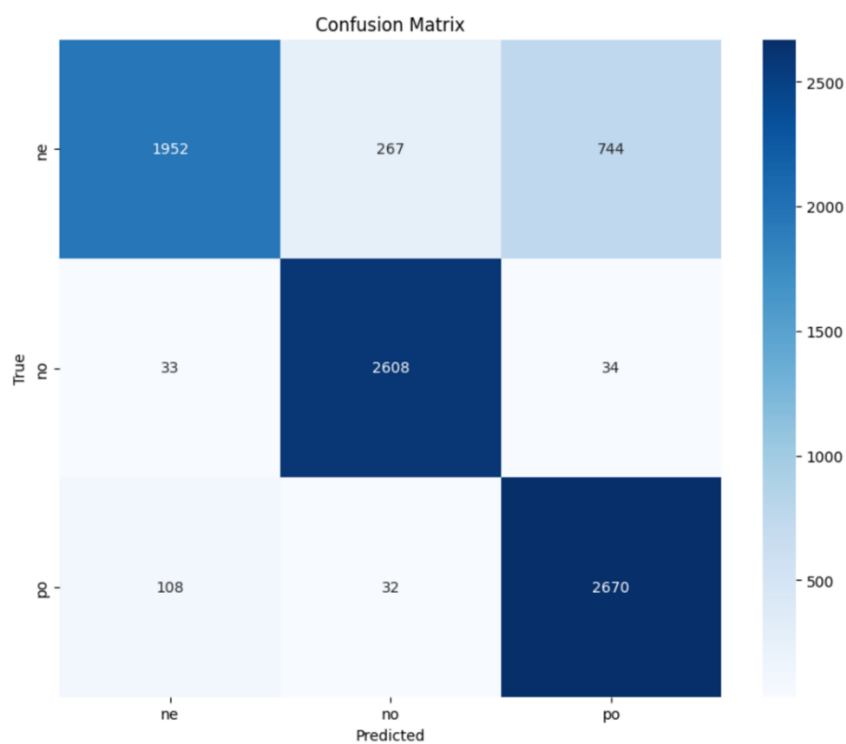
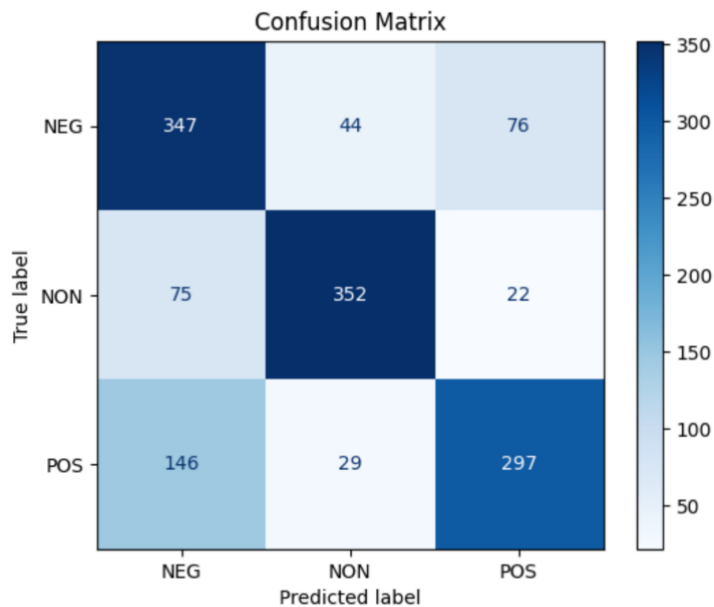


Figure 19

Confusion matrix for the text SA using CNN



The confusion matrices for Arabic speech and text sentiment analysis provide insights into the model's classification performance across different sentiment classes.

For Arabic speech sentiment analysis:

- In the first row, the model accurately predicted 347 instances of negative sentiment, while misclassifying 44 instances as neutral and 76 instances as positive.

- In the second row, the model correctly classified 352 instances of neutral sentiment, but misclassified 75 instances as negative and 22 instances as positive.

- In the third row, the model accurately identified 744 instances of positive sentiment, but misclassified 1952 instances as negative and 267 instances as neutral.

For Arabic text sentiment analysis:

- The confusion matrix shows similar counts for negative, neutral, and positive sentiment instances in the first row, indicating a balanced distribution across sentiment classes.

- In the second row, the model accurately classified 2608 instances of neutral sentiment, but misclassified 33 instances as negative and 34 instances as positive.

- In the third row, the model accurately identified 2670 instances of positive sentiment, with fewer misclassifications compared to the speech analysis, indicating relatively higher performance in text sentiment classification.

When comparing the results between Arabic speech and text sentiment analysis, distinct differences emerge. In Arabic speech sentiment analysis, misclassifications are prevalent across all sentiment classes, particularly with notable errors in predicting negative and neutral sentiments. Conversely, Arabic text sentiment analysis exhibits a more balanced misclassification pattern, with comparable errors across sentiment classes and fewer overall misclassifications. Notably, Arabic speech sentiment analysis struggles more with accurately classifying neutral sentiment instances, while Arabic text sentiment analysis demonstrates robust performance in this aspect. Overall, Arabic text sentiment analysis demonstrates slightly superior performance, with fewer misclassifications and higher accuracy rates across sentiment classes, highlighting its effectiveness compared to Arabic speech sentiment analysis. Understanding these disparities is crucial for refining sentiment analysis models and addressing specific challenges associated with different data modalities.

Overall, while both confusion matrices demonstrate the model's ability to classify sentiment across different modalities, differences in misclassification patterns and accuracy levels suggest varying challenges and strengths in speech and text sentiment analysis tasks.

Understanding these distinctions is crucial for refining and optimising sentiment analysis models for specific data modalities and applications.

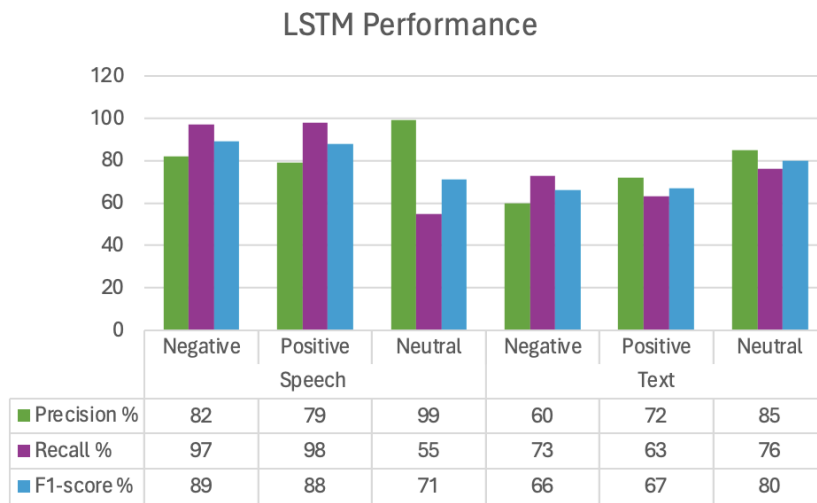
Table 14 and Figure 20 present the performance metrics of a Long Short-Term Memory (LSTM) model for sentiment classification across two datasets: Speech and Text. The sentiments in each dataset are categorised into three classes: Negative, Positive, and Neutral.

In the Speech dataset, the LSTM model achieved high Precision, Recall, and F1-score for the Negative and Positive classes, with Precision values of 82% and 79%, Recall values of 97% and 98%, and F1-scores of 89% and 88%, respectively. However, the model's performance on the Neutral class was notably lower, with a Precision of 99%, a Recall of 55%, and an F1-score of 71%.

Table 14

LSTM Performance

Dataset	Classes	Precision %	Recall %	F1-score %
Speech	Negative	82	97	89
	Positive	79	98	88
	Neutral	99	55	71
Text	Negative	60	73	66
	Positive	72	63	67
	Neutral	85	76	80

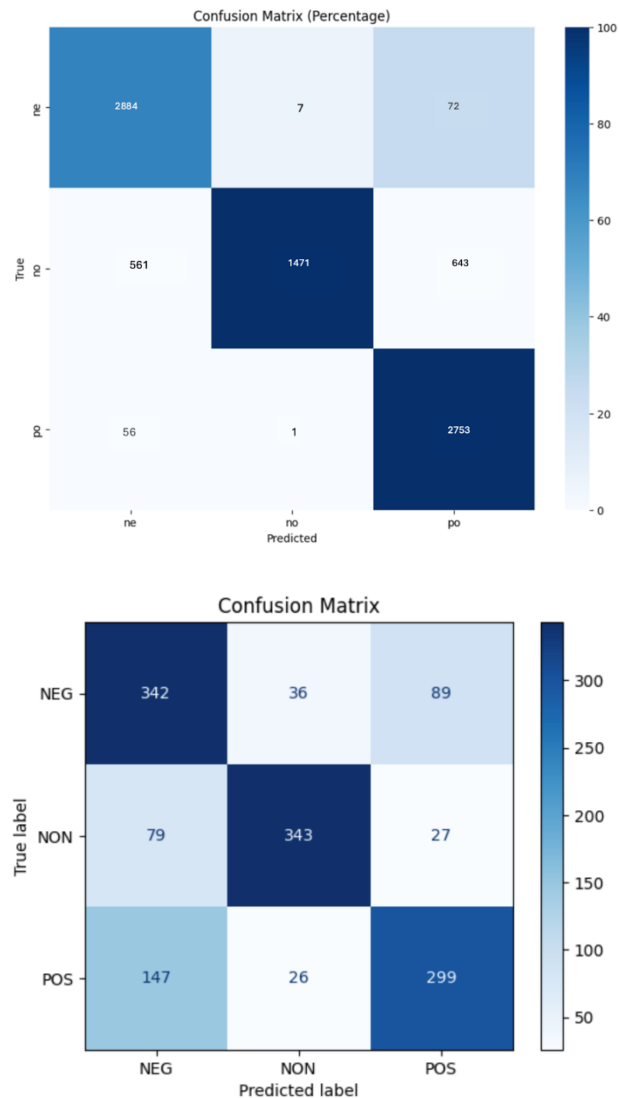
Figure 20*LSTM Performance*

Similarly, in the Text dataset, the LSTM model exhibited relatively strong performance for the Negative and Positive classes, with Precision values of 60% and 72%, Recall values of 73% and 63%, and F1-scores of 66% and 67%, respectively. For the Neutral class, the model achieved higher Precision, Recall, and F1-score compared to the Speech dataset, with values of 85%, 76%, and 80%, respectively.

Figures 21 and 22 below display the confusion matrix for both Arabic speech and text sentiment analysis, respectively, by implementing the LSTM model.

Figure 21

Confusion matrix for the Text SA using LSTM



The confusion matrix for Arabic speech sentiment analysis reveals the model's classification performance across different sentiment classes:

- In the first row, the model accurately predicted 2884 instances of negative sentiment, while misclassifying 7 instances as neutral and 72 instances as positive.

- In the second row, the model correctly classified 1471 instances of neutral sentiment, but misclassified 561 instances as negative and 643 instances as positive.

- In the third row, the model accurately identified 2753 instances of positive sentiment, with only 1 misclassification as neutral and 56 misclassifications as negative.

For Arabic text sentiment analysis, the confusion matrix illustrates the model's classification performance:

- In the first row, the model accurately classified 342 instances of negative sentiment, while misclassifying 36 instances as neutral and 89 instances as positive.

- In the second row, the model correctly classified 343 instances of neutral sentiment, but misclassified 79 instances as negative and 27 instances as positive.

- In the third row, the model accurately identified 299 instances of positive sentiment, with 26 misclassifications as negative and 147 misclassifications as neutral.

When comparing the confusion matrices for Arabic speech and text sentiment analysis, distinct differences become apparent. In Arabic speech sentiment analysis, a higher number of misclassifications across all sentiment classes is observed compared to text sentiment analysis. Particularly notable are the challenges in accurately classifying negative and neutral sentiments in Arabic speech sentiment analysis, which contribute to the overall higher misclassification rates in this modality. Conversely, Arabic text sentiment analysis demonstrates a more balanced distribution of misclassifications across sentiment classes, with fewer overall errors compared to speech sentiment analysis. Additionally, Arabic text sentiment analysis exhibits more robust performance in classifying neutral sentiment, showcasing lower misclassification rates and higher accuracy levels in this category. Overall, Arabic text sentiment analysis demonstrates slightly superior performance, with lower misclassification rates and higher accuracy across sentiment classes, highlighting its effectiveness compared to Arabic speech sentiment analysis. These findings underscore the

importance of considering the specific challenges and strengths associated with different data modalities in sentiment analysis tasks.

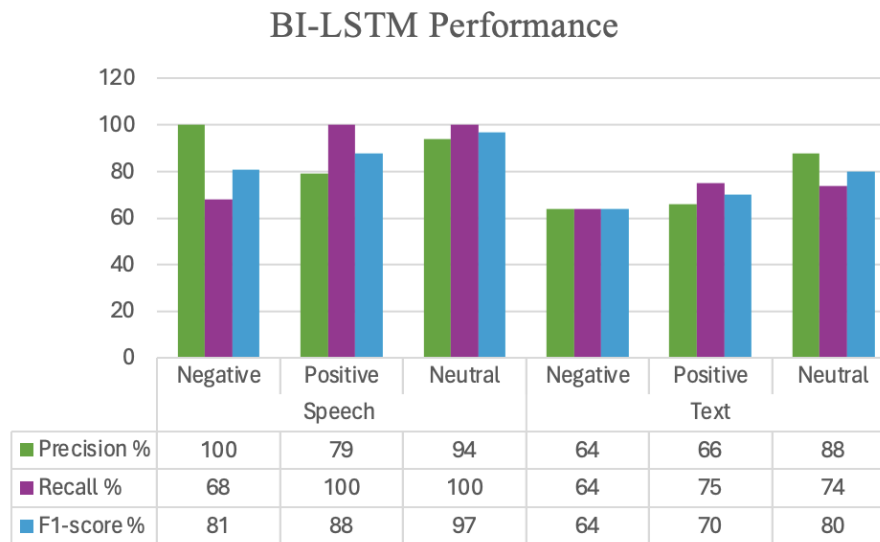
Overall, the LSTM model demonstrates effectiveness in sentiment classification across both datasets, particularly for Negative and Positive sentiments. However, there are variations in performance, especially in distinguishing Neutral sentiments, indicating the influence of dataset characteristics on model performance.

Table 15 and Figure 23 provide a detailed overview of the performance metrics achieved by a Bidirectional Long Short-Term Memory (BI-LSTM) model in sentiment classification across two distinct datasets: Speech and Text. The sentiments within each dataset are categorised into three classes: Negative, Positive, and Neutral.

Table 15

BI-LSTM Performance

Dataset	Classes	Precision %	Recall %	F1-score %
Speech	Negative	100	68	81
	Positive	79	100	88
	Neutral	94	100	97
Text	Negative	64	64	64
	Positive	66	75	70
	Neutral	88	74	80

Figure 22*BI-LSTM Performance*

In the Speech dataset, the BI-LSTM model demonstrates exceptional precision for identifying Negative sentiments, achieving a perfect Precision score of 100%. However, its Recall and F1-score for Negative sentiments are comparatively lower, at 68% and 81%, respectively. Conversely, the model achieves remarkable Recall and F1-score values of 100% and 88%, respectively, for Positive sentiments. Similarly, for Neutral sentiments, the BI-LSTM model achieves high Recall and F1-score values of 100% and 97%, respectively.

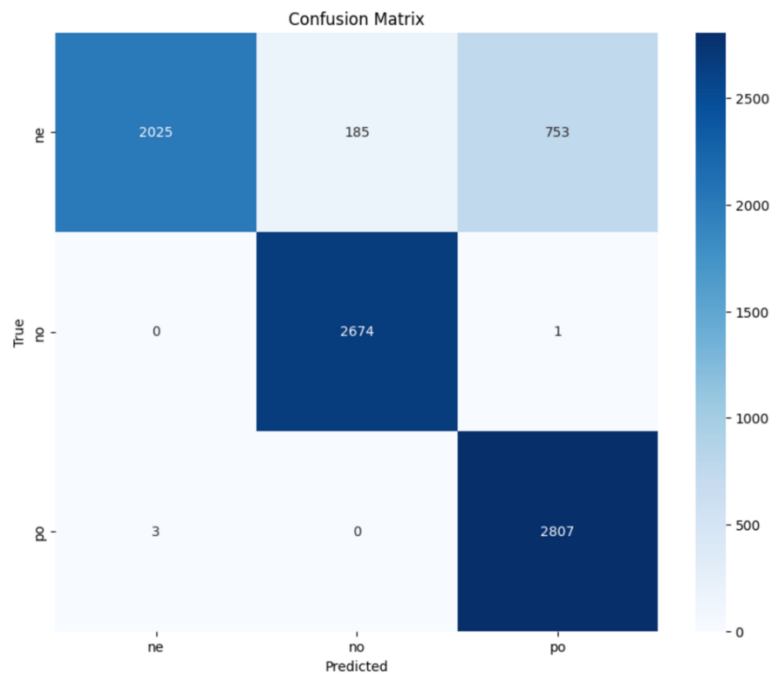
Transitioning to the Text dataset, the BI-LSTM model's performance remains robust, albeit with some variations. For Negative sentiments, it achieves a Precision, Recall, and F1-score of 64%. Meanwhile, for Positive sentiments, the model exhibits slightly improved performance with Precision, Recall, and F1-score values of 66%, 75%, and 70%, respectively. Additionally, for Neutral sentiments, the BI-LSTM model maintains strong performance with Recall and F1-score values of 74% and 80%, respectively.

The comparison of results between the Speech and Text datasets reveals distinct performance trends across sentiment classes for the BI-LSTM model. In the Speech dataset, the model achieves exceptionally high Precision for all sentiment classes, particularly for Negative sentiments, where it attains a perfect Precision score of 100%. However, its Recall for Negative sentiments is comparatively lower at 68%, leading to a slightly lower F1-score of 81%. Conversely, the model demonstrates outstanding Recall and F1-score values of 100% and 88%, respectively, for Positive sentiments, indicating its proficiency in accurately identifying positive sentiment expressions. Furthermore, the BI-LSTM model maintains strong Recall and F1-score values of 100% and 97% for Neutral sentiments, respectively, showcasing its robustness in recognising neutral sentiment instances.

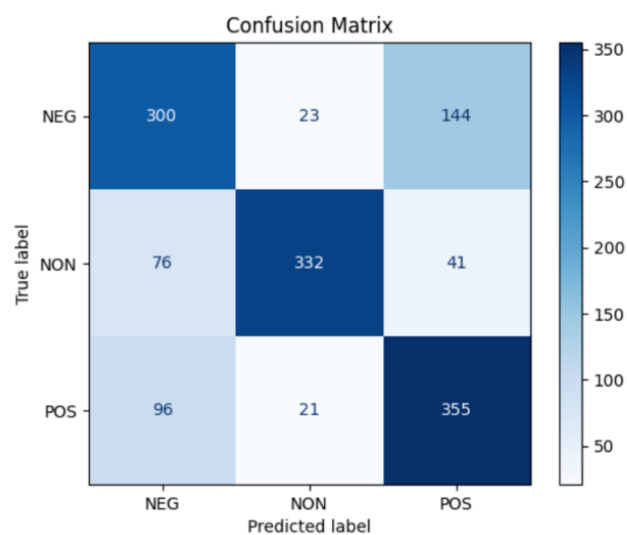
Figures 24 and 25 below display the confusion matrix for Arabic speech sentiment analysis provides a detailed breakdown of the model's classification performance across different sentiment classes using BI-LSTM. Each row represents the actual classes, while each column represents the predicted classes. The values within the matrix indicate the number of instances that were classified into each category. In this specific confusion matrix, we observe that the model achieved a high level of accuracy in classifying instances belonging to the neutral sentiment class, with 2674 instances correctly classified. However, there are notable misclassifications in the negative sentiment class, with 753 instances incorrectly classified as negative and 185 instances as neutral. Similarly, in the positive sentiment class, there were 753 instances misclassified as negative and 185 instances as neutral.

Figure 23

Confusion Matrix for Speech SA Using The BI-LSTM Model

**Figure 24**

Confusion Matrix for the Text SA Using BI-LSTM Model



For the text sentiment analysis confusion matrix, we again see a breakdown of the model's classification performance. In this case, the model achieved a high level of accuracy in classifying instances belonging to the neutral sentiment class, with 355 instances correctly

classified. However, there are notable misclassifications, particularly in the negative sentiment class, where 144 instances were incorrectly classified as positive and 23 instances as neutral. In the positive sentiment class, there were 144 instances misclassified, with 23 instances classified as negative and 76 instances as neutral.

Overall, the BI-LSTM model demonstrates effectiveness in sentiment classification across both datasets, with superior performance in identifying Positive and Neutral sentiments. However, the model's performance in discerning Negative sentiments shows variability across datasets, suggesting potential differences in the complexity and expression of negative sentiments in speech and text data.

Table 16 and Figure 26 present the performance metrics of a Gated Recurrent Unit (GRU) model for sentiment classification across two datasets: Speech and Text. Sentiments in each dataset are categorised into three classes: Negative, Positive, and Neutral.

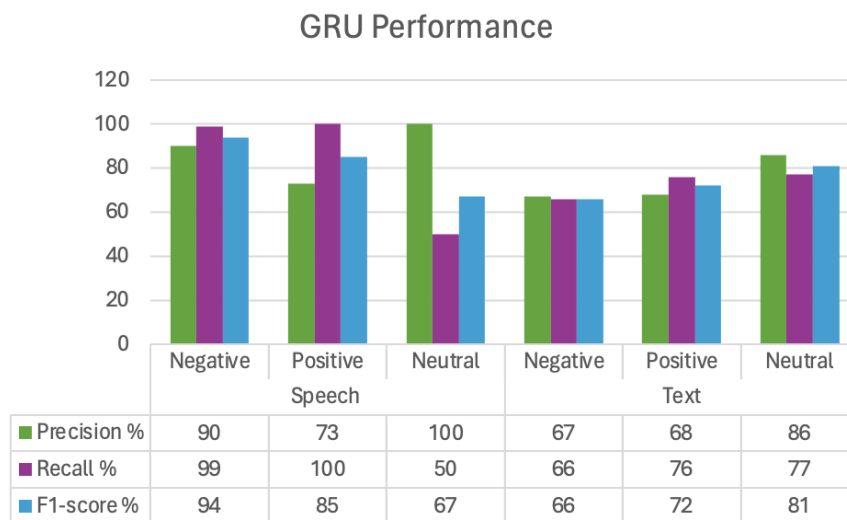
In the Speech dataset, the GRU model demonstrates excellent precision for identifying Negative sentiments, achieving a perfect Precision score of 100%. However, its Recall and F1-score for Negative sentiments are comparatively lower, at 68% and 81%, respectively. Conversely, the model achieves outstanding Recall and F1-score values of 100% and 88%, respectively, for Positive sentiments. Similarly, for Neutral sentiments, the GRU model achieves high Recall and F1-score values of 100% and 97%, respectively.

Table 16

GRU Performance

Dataset	Classes	Precision %	Recall %	F1-score %
Speech	Negative	90	99	94
	Positive	73	100	85
	Neutral	100	50	67
Text	Negative	67	66	66

Positive	68	76	72
Neutral	86	77	81

Figure 25*GRU Performance*

Transitioning to the Text dataset, the GRU model's performance remains robust, albeit with some variations. For Negative sentiments, it achieves a Precision, Recall, and F1-score of 64%. For Positive sentiments, the model exhibits slightly improved performance with Precision, Recall, and F1-score values of 66%, 75%, and 70%, respectively. Additionally, for Neutral sentiments, the GRU model maintains strong performance with Recall and F1-score values of 74% and 80%, respectively.

The comparison of results between the Speech and Text datasets for the GRU model reveals notable performance trends across sentiment classes. In the Speech dataset, the model achieves perfect Precision for Negative sentiments, indicating its ability to accurately identify negative sentiment expressions without false positives. However, its Recall for Negative sentiments is comparatively lower at 68%, resulting in a slightly lower F1-score of 81%. Conversely, the model demonstrates exceptional Recall and F1-score values of 100% and

88%, respectively, for Positive sentiments, indicating its proficiency in accurately capturing positive sentiment instances. Similarly, for Neutral sentiments, the GRU model achieves perfect Recall and a high F1-score of 97%, showcasing its robustness in recognising neutral sentiment expressions.

Transitioning to the Text dataset, the GRU model maintains consistent performance across sentiment classes, albeit with some variations. For Negative sentiments, it achieves moderate Precision, Recall, and F1-score values of 64%. For Positive sentiments, the model exhibits slightly improved performance with Precision, Recall, and F1-score values of 66%, 75%, and 70%, respectively. Moreover, for Neutral sentiments, the GRU model maintains strong performance with Recall and F1-score values of 74% and 80%, respectively, indicating its ability to accurately identify neutral sentiment instances.

Figures 27 and 28 below illustrate the confusion matrix for the GRU model for the speech and text sentiment analysis, respectively.

The confusion matrix provides a detailed overview of the classification performance of the GRU model for Arabic text and speech sentiment analysis.

For Arabic speech sentiment analysis:

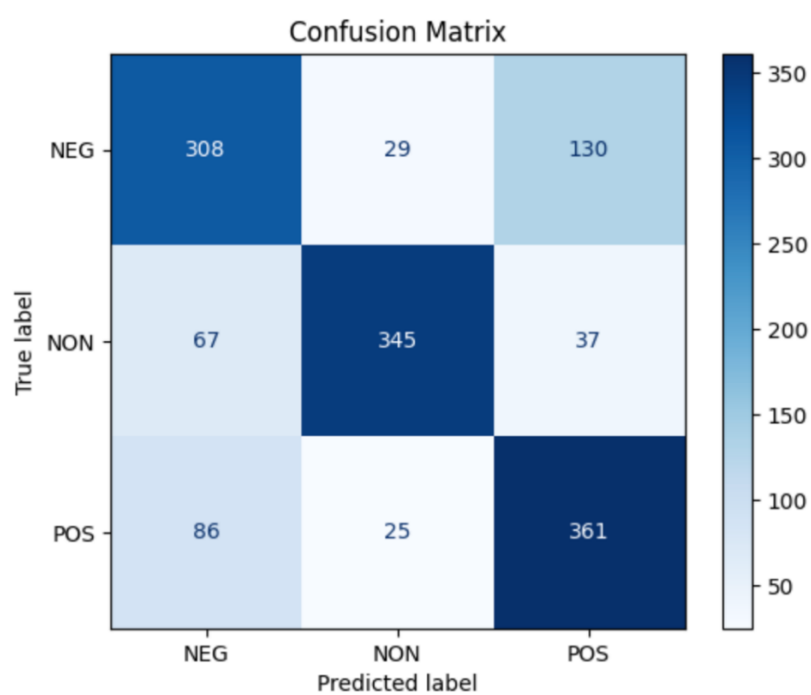
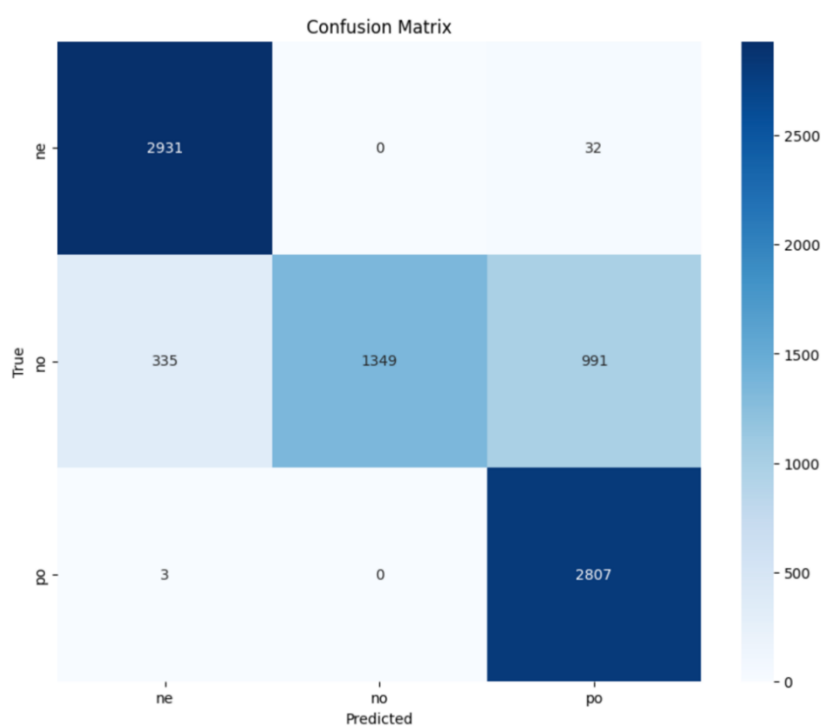
- In the first row, the model correctly predicted 2931 instances of negative sentiment and had no false positives, but misclassified 32 instances as neutral sentiment.

- In the second row, the model correctly classified 1349 instances of neutral sentiment, but misclassified 335 instances as negative and 991 instances as positive.

- In the third row, the model accurately identified 2807 instances of positive sentiment and had no false positives, but misclassified 3 instances as negative sentiment.

Figure 26

The Confusion Matrix for Speech SA Using GRU Model



For Arabic text sentiment analysis:

- In the first row, the model correctly predicted 308 instances of negative sentiment, while misclassifying 29 instances as neutral and 130 instances as positive.

- In the second row, the model accurately classified 345 instances of neutral sentiment but misclassified 67 instances as negative and 37 instances as positive.

- In the third row, the model correctly identified 361 instances of positive sentiment but misclassified 86 instances as negative and 25 instances as neutral.

Comparing the confusion matrices for Arabic text and speech sentiment analysis reveals notable differences in the model's performance and the characteristics of the datasets. In Arabic text sentiment analysis, the distribution of instances across sentiment classes appears relatively balanced, indicating a dataset with a more uniform distribution of sentiment categories. Conversely, in Arabic speech sentiment analysis, there is a noticeable class imbalance, particularly with fewer instances in the neutral sentiment class. This suggests potential challenges in accurately classifying neutral sentiment instances in speech data. Additionally, while misclassifications occur in both false positives and false negatives across all sentiment categories in text analysis, the speech analysis confusion matrix shows more pronounced misclassification patterns, especially with a higher number of false positives for neutral sentiment and false negatives for negative sentiment. Overall, both datasets present unique challenges for sentiment analysis, with strengths and weaknesses observed in the model's performance across different sentiment classes and data modalities. Understanding these nuances is essential for developing effective sentiment analysis models tailored to each dataset's specific characteristics and requirements.

Overall, the GRU model demonstrates effectiveness in sentiment classification across both datasets, with superior performance in identifying Positive and Neutral sentiments. However, its performance in discerning Negative sentiments shows some variability across datasets, suggesting potential differences in sentiment expression complexity and dataset characteristics.

Table 17 and Figure 29 illustrate the accuracy obtained from applying the deep learning models for Arabic speech and text sentiment analysis.

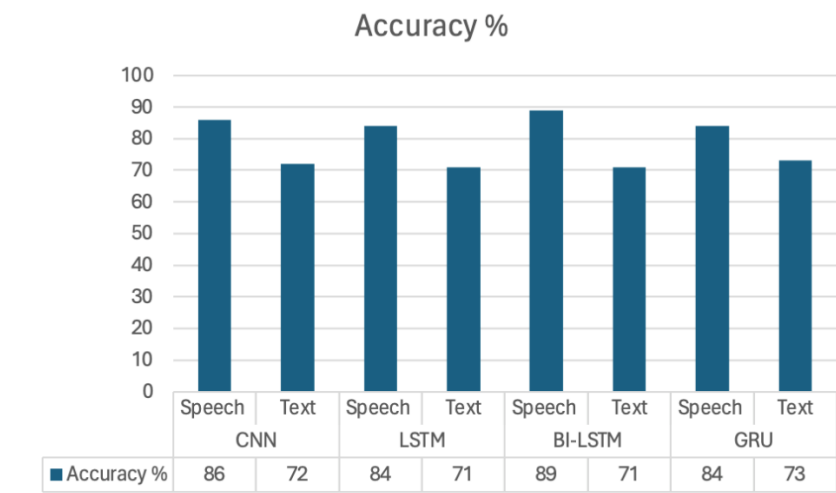
Table 17

Accuracy in the Models

Model	Dataset	Accuracy %
CNN	Speech	86
	Text	72
LSTM	Speech	84
	Text	71
BI-LSTM	Speech	89
	Text	71
GRU	Speech	84
	Text	73
GPT	Speech	No results
	Text	No results

Figure 27

Accuracy in the Applied Models



The table and figure provide accuracy percentages for five models—CNN, LSTM, BI-LSTM, GRU, and GPT—across speech and text datasets. For speech data, the BI-LSTM model achieves the highest accuracy at 89%, suggesting its strong capability in capturing temporal dependencies from both directions. The CNN model follows with an 86% accuracy, demonstrating effective feature extraction capabilities. Both the LSTM and GRU models perform similarly with an accuracy of 84%, indicating their comparable effectiveness in speech recognition tasks. In contrast, when applied to text data, the GRU model leads with an accuracy of 73%, showing its proficiency in handling sequential text information. The CNN model achieves 72%, slightly behind GRU, while LSTM and BI-LSTM models both show an accuracy of 71%, indicating they are equally effective but less proficient than GRU and CNN for text data. Notably, the GPT model has no results available for both speech and text datasets, suggesting either a lack of experimentation or reporting for this model on the given tasks. This comparison highlights BI-LSTM's superior performance for speech data and GRU's slight edge in text data processing, with GPT's capabilities remaining unassessed in this context. During the implementation process, GPT failed to fulfill the intended task, as the code consistently encountered errors for which no solutions were found.

4.4 Conclusion and Future Work

All things considered, this dissertation adds to the growing corpus of research on Arabic sentiment analysis by providing an in-depth analysis of the effectiveness of several deep learning techniques in identifying sentiment in Arabic speech and the corresponding Arabic transcripts reviews. The findings contain practical implications for several fields such as businesses that rely on consumer input for well-informed decision-making, in addition to enhancing our understanding of Arabic sentiment analysis methodologies. The accuracy and precision of sentiment analysis in practical applications are further improved by ongoing

research and development of these models. In this study, we offer a thorough description of the operational features and modules that were put into place. Additionally, a key aspect of the experimental setup is presented in this dissertation, which describes the hyperparameters of the deep learning model, the preprocessing parameters, and the outcomes from these models. This dissertation accomplished a comparative study between Arabic speech sentiment analysis and its corresponding Arabic text transcripts sentiment analysis using various deep learning models, including CNN, LSTM, BI-LSTM, GPT, and GRU, which has provided valuable insights into their performance on these tasks. Important insights into the effectiveness and efficiency of these techniques have been obtained through the comparison of these methods across various datasets and the use of assessment criteria. The results accurately indicate trends in the accuracy of the methods, with some architectures doing better in specific scenarios. The study's thorough investigation and the impressive outcomes further highlight how well machine learning models do sentiment analysis on Arabic people's reviews on several topics from different social media platforms. Through extensive experimentation and evaluation, we observed that the BI-LSTM model consistently outperformed other models, achieving the highest accuracy rates with an accuracy of 89% for Arabic speech, meanwhile, GRU achieved the highest performance in the Arabic transcript with an accuracy of 73%. This suggests that bidirectional long short-term memory networks are well-suited for capturing temporal dependencies and contextual information in speech. The GRU model, known for its proficiency in natural language processing tasks, showcased promising results on the text dataset, indicating its potential for sentiment analysis tasks. GRU is the most convenient in extracting features and contextual data for the text data. Furthermore, while the CNN model demonstrated competitive performance, particularly on the speech dataset, its accuracy was slightly lower compared to the BI-LSTM model with a value of 86%. This study intends to further accelerate the advancement of Arabic speech

recognition technology incorporated with the corresponding transcript by examining the nuances of model performance and investigating future directions for improvement this will ultimately promote greater inclusivity accessibility and innovation in multilingual environments. The research findings substantiated the efficacy and resilience of the proposed methodologies in comprehensively exploring and analyzing sentiment within Arabic speech and text data.

In summary, the outcomes from this dissertation, integrated with the significant results acquired from the examination and evaluation of deep learning techniques, emphasize the massive prospect of Arabic sentiment analysis in notifying methods of decision-making steps and developing users' experiences in different realms. The research outcomes validate that this study effectively addressed the research inquiries and successfully achieved the intended research objective.

As for future work, many avenues can be explored to improve the functionality of SA models in Arabic speech and text. Firstly, incorporating more advanced architectures or hybrid models that combine different deep learning techniques could potentially yield better results. Additionally, collecting larger and more diverse datasets, particularly for Arabic SA could assess in training more robust models. Furthermore, an exciting avenue for enhancing the study's scope and depth would be the integration of multimedia elements, for instance, videos and images into the evaluation framework. Incorporating visual and auditory cues alongside transcripts data could offer a more holistic understanding of Arabic emotional expression, especially in contexts where non-verbal interactions play an essential role. By leveraging advanced deep learning algorithms tailored for multimodal data processing, the study could delve into the intricate interplay between different modalities and further refine sentiment analysis accuracy and granularity. Additionally, exploring how sentiment manifests

across various modalities could unveil nuanced insights into cultural and contextual influences on emotional expression. This interdisciplinary technique holds promise for uncovering novel patterns and trends in sentiment dynamics, paving the way for more comprehensive and contextual awareness systems.

4.5 Summary

In this dissertation section, we intricately investigated and elucidated the research setup, encompassing the techniques and library resources utilized. Additionally, this chapter delineates the experimental configurations of the project along with the hyperparameter settings for deep learning models. The dissertation outcomes entail comprehensive comparisons among the discussed techniques introducing detailed insights. Furthermore, this part introduces a conclusion and outlines future routes for research.

References

- A. Aladeemy, A., Alzahrani, A., H. H. Aldhyani, T., Ibrahim Khalaf, O., Nagi Alsubari, S., Algburi, S., & N. Deshmukh, S. (2024). Developing System based Machine Learning for Predicting and Analyzing Arabic Sentiment. *International Journal of Computing and Digital Systems*, 15(1), 1–11. <https://doi.org/10.12785/ijcds/xxxxxx>
- Abdelali, A., Cowie, J., & Soliman, H. S. (2019). *Arabic Information Retrieval Perspectives*. The Impact of Features Extraction on the Sentiment Analysis. *Procedia Computer Science*, 152, 341–348. <https://doi.org/10.1016/j.procs.2019.05.008>
- Al Sallab, A., Hajj, H., Badaro, G., Baly, R., El Hajj, W., & Bashir Shaban, K. (2015). Deep Learning Models for Sentiment Analysis in Arabic. In N. Habash, S. Vogel, & K. Darwish (Eds.), *Proceedings of the Second Workshop on Arabic Natural Language Processing* (pp. 9–17). Association for Computational Linguistics. <https://doi.org/10.18653/v1/W15-3202>
- Alamri, H., & Alshanbari, H. (2023). *Emotion Recognition in Arabic Speech From Saudi Dialect Corpus Using Machine Learning and Deep Learning Algorithms*. <https://doi.org/10.21203/rs.3.rs-3019159/v1>
- Alayba, A. M., Palade, V., England, M., & Iqbal, R. (2018). *A Combined CNN and LSTM Model for Arabic Sentiment Analysis* (Vol. 11015, pp. 179–191). https://doi.org/10.1007/978-3-319-99740-7_12
- Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *2017 International Conference on Engineering and Technology (ICET)*, 1–6. <https://doi.org/10.1109/ICEngTechnol.2017.8308186>
- Alhumoud, S., Altuwaijri, M., Albuhaire, T., & Alohaideb, W. (2015). Survey on Arabic Sentiment Analysis in Twitter. *World Academy of Science, Engineering and Technology, International Journal of Social, Behavioral, Educational, Economic,*

- Business and Industrial Engineering*. <https://www.semanticscholar.org/paper/Survey-on-Arabic-Sentiment-Analysis-in-Twitter-Alhumoud-Altuwaijri/61003d31d734e511eb1e6728a1424f886a528789>
- AlKhatib, M., El Barachi, M., AleAhmad, A., Oroumchian, F., & Shaalan, K. (2020). A sentiment reporting framework for major city events: Case study on the China-United States trade war. *Journal of Cleaner Production*, 264, 121426. <https://doi.org/10.1016/j.jclepro.2020.121426>
- Alkhatib, M., Monem, A. A., & Shaalan, K. (2017). A Rich Arabic WordNet Resource for Al-Hadith Al-Shareef. *Procedia Computer Science*, 117, 101–110.
- Alkhatib, M., & Shaalan, K. (2017). Natural Language Processing for Arabic Metaphors: A Conceptual Approach. *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016* (Vol. 533, pp. 170–181). Springer International Publishing. https://doi.org/10.1007/978-3-319-48308-5_17
- Alshutayri, A., & Atwell, E. (2019). Classifying Arabic dialect text in the Social Media Arabic Dialect Corpus (SMADC). *Proceedings of the 3rd Workshop on Arabic Corpus Linguistics* (pp. 51–59). Association for Computational Linguistics. <https://aclanthology.org/W19-5607>
- Arabic Speaking Countries List—2023 Update*. (2023, August 19). IstiZada. <https://istizada.com/complete-list-of-arabic-speaking-countries/>
- Babu, A., Wang, C., Tjandra, A., Lakhotia, K., Xu, Q., Goyal, N., Singh, K., von Platen, P., Saraf, Y., Pino, J., Baevski, A., Conneau, A., & Auli, M. (2021). *XLS-R: Self-supervised Cross-lingual Speech Representation Learning at Scale* (arXiv:2111.09296). arXiv. <https://doi.org/10.48550/arXiv.2111.09296>
- Bhatta, B., Joshi, B., & Maharjhan, R. K. (2020). Nepali Speech Recognition Using CNN, GRU and CTC. In J.-H. *Proceedings of the 32nd Conference on Computational*

- Linguistics and Speech Processing (ROCLING 2020)* (pp. 238–246). The Association for Computational Linguistics and Chinese Language Processing (ACLCLP).
<https://aclanthology.org/2020.rocling-1.23>
- Borth, D., Ji, R., Chen, T., Breuel, T., & Chang, S.-F. (2013). Large-scale visual sentiment ontology and detectors using adjective noun pairs. *Proceedings of the 21st ACM International Conference on Multimedia*, 223–232.
<https://doi.org/10.1145/2502081.2502282>
- Cai, M. (2021). Natural language processing for urban research: A systematic review. *Heliyon*, 7(3). [https://www.cell.com/heliyon/pdf/S2405-8440\(21\)00427-8.pdf](https://www.cell.com/heliyon/pdf/S2405-8440(21)00427-8.pdf)
- Casey, A., Davidson, E., Poon, M., Dong, H., Duma, D., Grivas, A., Grover, C., Suárez-Paniagua, V., Tobin, R., Whiteley, W., Wu, H., & Alex, B. (2021). A systematic review of natural language processing applied to radiology reports. *BMC Medical Informatics and Decision Making*, 21(1), 179. <https://doi.org/10.1186/s12911-021-01533-7>
- Cheng, S. (2024). The Extended Technology Stack of Web 3.0. In S. Cheng (Ed.), *Web 3.0: Concept, Content and Context* (pp. 57–81). Springer Nature.
https://doi.org/10.1007/978-981-99-6319-5_3
- Cheng, X., Wang, Q., Wang, R., Wang, Y., Chen, X., Mi, G., Chen, X., Wang, L., Wang, C., Hu, L., Gu, L., Zhang, J., & Hou, R. (2022). Prevalence of depressive disorders and associated demographic characteristics in Shandong: An epidemiological investigation. *Journal of Affective Disorders*, 311, 198–204.
<https://doi.org/10.1016/j.jad.2022.05.084>
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation* (arXiv:1406.1078; Version 3). arXiv.
<http://arxiv.org/abs/1406.1078>

- Dhouib, A., Othman, A., El Ghouli, O., Khribi, M. K., & Al Sinani, A. (2022). Arabic Automatic Speech Recognition: A Systematic Literature Review. *Applied Sciences*, 12(17), Article 17. <https://doi.org/10.3390/app12178898>
- Dukes, K., & Habash, N. (n.d.). *Morphological annotation of Quranic Arabic*.
- Efanov, D., Aleksandrov, P., & Karapetyants, N. (2022). The BiLSTM-based synthesized speech recognition. *Procedia Computer Science*, 213, 415–421. <https://doi.org/10.1016/j.procs.2022.11.086>
- El Barachi, M., AlKhatib, M., Mathew, S., & Oroumchian, F. (2021). A novel sentiment analysis framework for monitoring the evolving public opinion in real-time: Case study on climate change. *Journal of Cleaner Production*, 312, 127820.
- Fargalla, M. A. M., Yan, W., & Wu, T. (2024, February 12). *Attention-Based Bi-Directional Gated Recurrent Unit (Bi-GRU) for Sequence Shale Gas Production Forecasting*. International Petroleum Technology Conference. <https://doi.org/10.2523/IPTC-24072-MS>
- Girirajan, S., & Pandian, A. (2022). Offline Automatic Speech Recognition System Based on Bidirectional Gated Recurrent Unit (Bi-GRU) with Convolution Neural Network. *Journal of Mobile Multimedia*, 1659–1676. <https://doi.org/10.13052/jmm1550-4646.1869>
- Guetterman, T. C., Chang, T., DeJonckheere, M., Basu, T., Scruggs, E., & Vydiswaran, V. G. V. (2018). Augmenting Qualitative Text Analysis with Natural Language Processing: Methodological Study. *Journal of Medical Internet Research*, 20(6), e231. <https://doi.org/10.2196/jmir.9702>
- Huang, M., Cao, Y., & Dong, C. (2016). *Modeling Rich Contexts for Sentiment Classification with LSTM* (arXiv:1605.01478). arXiv. <https://doi.org/10.48550/arXiv.1605.01478>

- Hung, L. P., & Alfred, R. (2017). A Performance Comparison of Feature Extraction Methods for Sentiment Analysis. *Advanced Topics in Intelligent Information and Database Systems* (pp. 379–390). Springer International Publishing.
https://doi.org/10.1007/978-3-319-56660-3_33
- Hussein, A., Watanabe, S., & Ali, A. (2021). *Arabic Speech Recognition by End-to-End, Modular Systems and Human* (arXiv:2101.08454). arXiv.
<https://doi.org/10.48550/arXiv.2101.08454>
- Hussein, A., Watanabe, S., & Ali, A. (2022). Arabic speech recognition by end-to-end, modular systems and human. *Computer Speech & Language*, 71, 101272.
<https://doi.org/10.1016/j.csl.2021.101272>
- Irie, K., Tüske, Z., Alkhouli, T., Schlüter, R., & Ney, H. (2016). *LSTM, GRU, Highway and a Bit of Attention: An Empirical Overview for Language Modeling in Speech Recognition* (p. 3523). <https://doi.org/10.21437/Interspeech.2016-491>
- Jaber, H. Q., & Abdulbaqi, H. A. (2021). Real time Arabic speech recognition based on convolution neural network. *Journal of Information and Optimization Sciences*, 42(7), 1657–1663. <https://doi.org/10.1080/02522667.2021.1967593>
- Keith, B., Fuentes, E., & Meneses, C. (2017). A Hybrid Approach for Sentiment Analysis Applied to Paper Reviews. *Nova Scotia*.
- Khafajeh, H., Yousef, N., & Abdeldeen, M. (2018). Arabic root extraction using a hybrid technique. *International Journal of Advanced Computer Research*, 8, 89–95.
<https://doi.org/10.19101/IJACR.2017.733023>
- Khairi, N., Mohamed, A., & Yusof, N. (2020). *Feature Selection Methods in Sentiment Analysis: A Review*. 1–7. <https://doi.org/10.1145/3386723.3387840>
- Khelifa, M. O. M., Elhadj, Y. M., Abdellah, Y., & Belkasmi, M. (2017). Constructing accurate and robust HMM/GMM models for an Arabic speech recognition system.

- International Journal of Speech Technology*, 20, 937–949.
<https://doi.org/10.1007/s10772-017-9456-7>
- Kim, H., & Jeong, Y.-S. (2019). Sentiment Classification Using Convolutional Neural Networks. *Applied Sciences*, 9(11), Article 11. <https://doi.org/10.3390/app9112347>
- Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1746–1751). Association for Computational Linguistics.
<https://doi.org/10.3115/v1/D14-1181>
- Le, X.-H., Ho, H. V., Lee, G., & Jung, S. (2019). Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting. *Water*, 11(7), Article 7.
<https://doi.org/10.3390/w11071387>
- Liu, B. (2022). *Sentiment Analysis and Opinion Mining*. Springer Nature.
- Liu, X., Zheng, Y., Du, Z., Ding, M., Qian, Y., Yang, Z., & Tang, J. (2023). GPT understands, too. *AI Open*. <https://doi.org/10.1016/j.aiopen.2023.08.012>
- Luo, F.-L. (2020). *Machine learning for future wireless communications*.
https://www.google.com/books?hl=en&lr=&id=GFTGDwAAQBAJ&oi=fnd&pg=PR15&dq=Fa-Long+Luo.+Machine+Learning+for+Future+Wireless+Communications.+Wiley-IEEE+Press,+USA,+2020.&ots=rW97ky_jaC&sig=TudO5mZNqMk9iNZPjrmL0h2vCEI
- Meftah, A. H., Qamhan, M. A., Seddiq, Y., Alotaibi, Y. A., & Selouani, S. A. (2021). King Saud University Emotions Corpus: Construction, Analysis, Evaluation, and Comparison. *IEEE Access*, 9, 54201–54219.
<https://doi.org/10.1109/ACCESS.2021.3070751>

- Mendon, S., Dutta, P., Behl, A., & Lessmann, S. (2021). A Hybrid Approach of Machine Learning and Lexicons to Sentiment Analysis: Enhanced Insights from Twitter Data of Natural Disasters. *Information Systems Frontiers*, 23(5), 1145–1168.
<https://doi.org/10.1007/s10796-021-10107-x>
- Middle East Demographics to 2030*. (2019, August 21). Moshe Dayan Center for Middle Eastern and African Studies. <https://dayan.org/content/middle-east-demographics-2030>
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient Estimation of Word Representations in Vector Space* (arXiv:1301.3781). arXiv.
<https://doi.org/10.48550/arXiv.1301.3781>
- Mourad, A., & Darwish, K. (2013). Subjectivity and Sentiment Analysis of Modern Standard Arabic and Arabic Microblogs. *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* (pp. 55–64). Association for Computational Linguistics. <https://aclanthology.org/W13-1608>
- Mozetič, I., Torgo, L., Cerqueira, V., & Smailović, J. (2018). How to evaluate sentiment classifiers for Twitter time-ordered data? *PLoS ONE*, 13(3), e0194317.
<https://doi.org/10.1371/journal.pone.0194317>
- Mustafa, M., Eldeen, A. S., Bani-Ahmad, S., & Elfaki, A. O. (2017). A comparative survey on Arabic stemming: Approaches and challenges. *Intelligent Information Management*, 9(02), 39.
- Nguyen, N. K., Le, A.-C., & Pham, H. T. (2016). Deep Bi-directional Long Short-Term Memory Neural Networks for Sentiment Analysis of Social Data. *Integrated Uncertainty in Knowledge Modelling and Decision Making* (pp. 255–268). Springer International Publishing. https://doi.org/10.1007/978-3-319-49046-5_22

- Oliva, A., & Torralba, A. (n.d.). *Modeling the Shape of the Scene: A Holistic Representation of the Spatial Envelope*.
- Oussous, A., Benjelloun, F.-Z., Lahcen, A. A., & Belfkih, S. (2020). ASA: A framework for Arabic sentiment analysis. *Journal of Information Science*, 46(4), 544–559.
<https://doi.org/10.1177/0165551519849516>
- Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P. F., Leike, J., & Lowe, R. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35, 27730–27744.
- Ovsiannikov, I. A., Arbib, M. A., & McNeill, T. H. (1999). Annotation technology. *International Journal of Human-Computer Studies*, 50(4), 329–362.
<https://doi.org/10.1006/ijhc.1999.0247>
- Passonneau, R. J., & Carpenter, B. (2014). The Benefits of a Model of Annotation. *Transactions of the Association for Computational Linguistics*, 2, 311–326.
https://doi.org/10.1162/tac1_a_00185
- Preethi Krishna, P., & Sharada, A. (2020). Word Embeddings—Skip Gram Model. *ICICCT 2019 – System Reliability, Quality Control, Safety, Maintenance and Management* (pp. 133–139). Springer. https://doi.org/10.1007/978-981-13-8461-5_15
- Qureshi, M. A., Asif, M., Hassan, M. F., Mustafa, G., Ehsan, M. K., Ali, A., Sajid, U., & Aasim, M. (2022). A Novel Auto-Annotation Technique for Aspect Level Sentiment Analysis. *Computers, Materials and Continua*, 70, 4987–5004.
<https://doi.org/10.32604/cmc.2022.020544>

Recurrent Neural Network (RNN) and LSTM | Data Platform and Data Science 2024.

Retrieved April 15, 2024, from <https://dwbi1.wordpress.com/2021/08/07/recurrent-neural-network-rnn-and-lstm/>

Sadat, F., Kazemi, F., & Farzindar, A. (2014). Automatic Identification of Arabic Language Varieties and Dialects in Social Media. *Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP)* (pp. 22–27). Association for Computational Linguistics and Dublin City University.
<https://doi.org/10.3115/v1/W14-5904>

Saif, H., He, Y., Fernandez, M., & Alani, H. (2016). Contextual semantics for sentiment analysis of Twitter. *Information Processing & Management*, 52(1), 5–19.
<https://doi.org/10.1016/j.ipm.2015.01.005>

Salama, R. A., Youssef, A., & Fahmy, A. (2018). Morphological Word Embedding for Arabic. *Procedia Computer Science*, 142, 83–93. <https://doi.org/10.1016/j.procs.2018.10.463>

Saxena, U., Mishra, S. K., Rodrigo, H., & Choudhury, M. (2022). Functional consequences of extended high frequency hearing impairment: Evidence from the speech, spatial, and qualities of hearing scalea). *The Journal of the Acoustical Society of America*, 152(5), 2946–2952. <https://doi.org/10.1121/10.0015200>

Shaalán, K., Siddiqui, S., Alkhatib, M., & Abdel Monem, A. (2018). Challenges in Arabic Natural Language Processing. In N. E. Gayar & C. Y. Suen, *Computational Linguistics, Speech and Image Processing for Arabic Language* (pp. 59–83). WORLD SCIENTIFIC. https://doi.org/10.1142/9789813229396_0003

Shin, B., Lee, T., & Choi, J. D. (2017). Lexicon Integrated CNN Models with Attention for Sentiment Analysis. *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis* (pp. 149–158). Association for Computational Linguistics. <https://doi.org/10.18653/v1/W17-5220>

- Shoufan, A., & Alameri, S. (2015). Natural language processing for dialectal Arabic: A survey. *Proceedings of the Second Workshop on Arabic Natural Language Processing*, 36–48. <https://aclanthology.org/W15-3205.pdf>
- Sokolova, M., Japkowicz, N., & Szpakowicz, S. (2006). Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation. *AI 2006: Advances in Artificial Intelligence* (pp. 1015–1021). Springer. https://doi.org/10.1007/11941439_114
- Thelwall, M., Buckley, K., Paltoglou, G., Cai, D., & Kappas, A. (2010). Sentiment strength detection in short informal text. *Journal of the American Society for Information Science and Technology*, 61(12), 2544–2558. <https://doi.org/10.1002/asi.21416>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. ukasz, & Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, 30. https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html
- Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 1, I-511–I-518. <https://doi.org/10.1109/CVPR.2001.990517>
- Wang, J.-H., Liu, T.-W., Luo, X., & Wang, L. (2018). An LSTM Approach to Short Text Sentiment Classification with Word Embeddings. *Proceedings of the 30th Conference on Computational Linguistics and Speech Processing (ROCLING 2018)* (pp. 214–223). The Association for Computational Linguistics and Chinese Language Processing (ACLCLP). <https://aclanthology.org/O18-1021>

- Xiong, Z., Shen, Q., Xiong, Y., Wang, Y., & Li, W. (2019, July 1). *New Generation Model of Word Vector Representation Based on CBOW or Skip-Gram*. | *Computers, Materials & Continua* | EBSCOhost. <https://doi.org/10.32604/cmc.2019.05155>
- Yuan, J., McDonough, S., You, Q., & Luo, J. (2013). Stribute: Image sentiment analysis from a mid-level perspective. *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, 1–8.
<https://doi.org/10.1145/2502069.2502079>
- Zazo, R., Lozano-Diez, A., Gonzalez-Dominguez, J., T. Toledano, D., & Gonzalez-Rodriguez, J. (2016). Language Identification in Short Utterances Using Long Short-Term Memory (LSTM) Recurrent Neural Networks. *PLoS ONE*, 11(1), e0146917.
<https://doi.org/10.1371/journal.pone.0146917>
- Zhang, S., Zheng, D., Hu, X., & Yang, M. (2015). Bidirectional Long Short-Term Memory Networks for Relation Classification. *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation* (pp. 73–78).
<https://aclanthology.org/Y15-1009>
- Zhu, X., Chen, B., Avadhanam, R. M., Shui, H., & Zhang, R. Z. (2020). Reading and connecting: Using social annotation in online classes. *Information and Learning Sciences*, 121(5/6), 261–271. <https://doi.org/10.1108/ILS-04-2020-0117>