

PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
FERHAT ABBAS UNIVERSITY SETIF-1
FACULTY OF SCIENCES
DEPARTMENT OF COMPUTER SCIENCE



PhD Dissertation

PRESENTED BY:

Mohamed BERRIMI

AS A REQUIREMENT FOR OBTAINING THE **Doctoral degree** IN COMPUTER SCIENCE
OPTION: SMART SYSTEMS AND MACHINE LEARNING

Deep models for Generating and Understanding Textual Arabic Data

DEFENDED ON: DECEMBER, 2ND 2023

PHD DEFENSE BOARD:

| | | |
|---------------------------|-----------------------------|---------------|
| Pr. Abdelouahab MOUSSAOUI | University of Setif | Chairman |
| Dr. Abdelaziz LAKHFIF | University of Setif | Examiner |
| Dr. Said GADRI | University of Msila | Examiner |
| Dr. Mohamed SAIDI | University of Setif | Supervisor |
| Dr. Mourad OUSSALAH | University of Oulu, Finland | Co-supervisor |

Abstract

The Arabic language is renowned for its rich linguistic heritage and complex morphology along with its wide range of dialects, which pose significant challenges to both research and developer communities. Indeed, spoken by over 400 million people globally and practiced by more than 2 billion people within the Muslim community, Arabic plays a vital role in fostering communication ecosystem and cultural norms in Arab world and beyond. Therefore, Arabic natural language processing (NLP) research is experiencing a growing interest to tackle distinctive and inherent challenges encountered by researchers when attempting to develop efficient learning models capable of comprehending, processing, and enabling basic NLP modules of Arabic language. The thesis is based on seven publications that make significant contributions to Arabic language understanding and generation, annotating Arabic corpora and training new models.

The thesis aims to address multiple challenges faced in the field of Arabic NLP by proposing novel deep learning contributions and exploring their applications across a range of tasks, particularly in sentiment analysis, hate speech detection, language model generation, speech emotion recognition, dialect identification, and document classification.

In this thesis, our primary focus lies in advancing the research efforts and making significant contributions to the Arabic language domain through the utilization of the state-of-the-art (SOTA) deep learning techniques. Another specific objective is to enhance Arabic resources by creating diverse Arabic text and speech corpora encompassing various dialects that are relevant to a variety of NLP tasks. In addition, we delve into the exploration and development of effective models specifically designed for Arabic language processing applied for the aforementioned NLP tasks. These models were tailored to perform consistently in both Modern Standard Arabic (MSA) and dialectal Arabic datasets.

By developing and evaluating these models and their applications, this thesis contributes sig-

nificantly to the field of Arabic NLP, paving the way for future research and advancements in solving the unique challenges of processing Arabic text and speech.

ملخص

تشتهر اللغة العربية بتراثها اللغوي الغني وتشكيلاتها المعقدة إلى جانب مجموعة واسعة من اللهجات ، مما يشكل تحديات كبيرة لكل من مجتمعات البحث والمطورين. في الواقع ، يتحدث بها أكثر من ٤٠٠ مليون شخص على مستوى العالم ويمارسها أكثر من مليار شخص داخل المجتمع الإسلامي ، تلعب اللغة العربية دورًا حيويًا في تعزيز النظام الإيكولوجي للتواصل والأعراف الثقافية في العالم العربي وخارجه. لذلك ، تشهد أبحاث معالجة اللغة العربية اهتمامًا متزايدًا بمعالجة التحديات المميزة والمتأصلة التي يواجهها الباحثون عند محاولة تطوير نماذج تعليمية فعالة قادرة على فهم ومعالجة وتمكين وحدات المعالجة اللغوية للغة العربية.

تهدف هذه أطروحة الدكتوراه إلى معالجة التحديات العديدة التي تواجهها في مجال معالجة اللغة الطبيعية للعربية من خلال اقتراح مساهمات جديدة في التعلم العميق واستكشاف تطبيقاتها في مجموعة من المهام، ولا سيما في مجالات تحليل المشاعر وكشف الخطاب الكراهي، ونماذج اللغة والتعرف على العواطف المتعلقة بالكلام، وتحديد اللهجة، وتصنيف الوثائق.

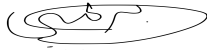
في هذه الأطروحة، يكمن التركيز الأساسي لدينا في تطوير جهود البحث وتقديم مساهمات هامة في مجال اللغة العربية من خلال استخدام تقنيات التعلم العميق الحديثة. هدفنا المحدد هو تعزيز الموارد العربية من خلال إنشاء مجموعات نصوص وكلام عربية متنوعة تشمل مختلف اللهجات التي يمكن تطبيقها في مجموعة متنوعة من المهام في معالجة اللغة الطبيعية، بما في ذلك تصنيف النصوص، وتحديد اللهجة، وتحليل المشاعر من النصوص والكلام، ونمذجة اللغة. بالإضافة إلى ذلك، نتناول استكشاف وتطوير نماذج فعالة مصممة خصيصًا لمعالجة اللغة العربية وتطبيقها في المهام المذكورة، بنية القيام تطورات مهمة في هذا المجال. من خلال تطوير وتقييم هذه النماذج الفعالة وتطبيقها، تسهم هذه الأطروحة بشكل كبير في مجال معالجة اللغة الطبيعية للعربية، مما يسهل الطريق للبحث المستقبلي والتقدم في حل التحديات الفريدة التي تواجه معالجة النصوص والكلام العربي.

Intellectual Property

I hereby declare that this work is the result of my own independent work, and that credit has been given to the work of others. This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgment.

© 2023 University of Ferhat Abbas 1,
Mohamed Berrimi

Signed

A handwritten signature in black ink, enclosed within a hand-drawn oval. The signature is stylized and appears to be the name 'Mohamed Berrimi'.

Acknowledgements

الحمد لله الذي هدانا لهذا وما كنا لنهتدي لولا ان هدانا الله

I have been on my own struggling through the years to accomplish this, it was never easy.

I wanna thank God and myself.

Contents

| | |
|--|-----------|
| 1 Introduction | 1 |
| 1.1 Context and Motivation | 1 |
| 1.2 Research Contributions | 2 |
| 1.2.1 Arabic text and speech analysis | 2 |
| 1.2.2 Arabic dialect identification: Challenges and Approaches | 4 |
| 1.2.3 Developing DziriBERT: A Pretrained Language Model for Algerian Dialect | 4 |
| 1.2.4 Arabic document classification | 5 |
| 1.3 Statement of personal contribution | 5 |
| 1.4 Role in Research: Challenges, Achievements, and Lessons | 5 |
| 1.5 Thesis organization | 6 |
| 1.5.1 Chapter 2: Background, SOTA, and related works | 7 |
| 1.5.2 Chapter 3: Arabic text and speech analysis | 7 |
| 1.5.3 Chapter 4: Arabic dialect identification | 8 |
| 1.5.4 Chapter 5: DziriBERT: a Pretrained language model for Algerian dialect | 8 |
| 1.5.5 Chapter 6: Effective deep learning models for Arabic text classification | 8 |
| 1.6 Publications | 9 |
| 1.6.1 International Journals | 9 |
| 1.6.2 International Conferences | 9 |
| 2 Background | 11 |
| 2.1 Arabic language in NLP, background and challenges | 11 |
| 2.2 Theoretical background: a systematic overview | 15 |
| 2.2.1 Data engineering and quality assurance | 16 |
| 2.2.2 Machine learning | 17 |

| | | |
|----------|--|-----------|
| 2.2.3 | Logistic Regression | 17 |
| 2.2.4 | Support Vector Machines | 17 |
| 2.3 | Deep Learning | 18 |
| 2.3.1 | LSTM | 18 |
| 2.3.2 | GRU | 19 |
| 2.3.3 | Convolutional neural networks | 19 |
| 2.3.4 | Attention Mechanism | 20 |
| 2.3.5 | Transformer model architecture | 21 |
| 2.4 | Word representations | 24 |
| 2.4.1 | Handcrafted features | 24 |
| 2.4.2 | Word2vec | 26 |
| 2.4.3 | GloVe | 27 |
| 2.4.4 | FastText embeddings | 28 |
| 2.4.5 | Contextual embeddings and Transfer learning | 28 |
| 2.5 | Related works | 31 |
| 2.5.1 | Related works in Arabic sentiment analysis | 31 |
| 2.5.2 | Related works in Arabic offensive speech detection | 34 |
| 2.5.3 | Related works in Arabic Emotion Speech Recognition | 36 |
| 2.5.4 | Related Works in Arabic Dialect Identification | 38 |
| 2.5.5 | Related works in Arabic Language Models | 40 |
| 2.5.6 | Related works in Text Classification | 43 |
| 2.6 | Conclusion | 46 |
| 3 | Arabic text and speech analysis | 47 |
| 3.1 | Arabic Sentiment Analysis | 47 |
| 3.1.1 | Introduction | 47 |
| 3.1.2 | Proposed system | 51 |
| 3.1.3 | Experimentation | 58 |
| 3.1.4 | Results | 62 |
| 3.1.5 | Statistical evaluation | 64 |
| 3.1.6 | Discussions and Implications | 65 |
| 3.1.7 | Conclusion | 69 |

| | | |
|----------|---|------------|
| 3.2 | SER for Algerian Dialect | 70 |
| 3.2.1 | Introduction | 70 |
| 3.2.2 | Methodology | 72 |
| 3.2.3 | Baseline models | 76 |
| 3.2.4 | Proposed DeepEmoNet model | 76 |
| 3.2.5 | Experiments and results | 76 |
| 3.2.6 | Discussion | 78 |
| 3.2.7 | Conclusion and future work | 80 |
| 3.3 | Inappropriate speech detection in Arabic text | 81 |
| 3.3.1 | Introduction | 81 |
| 3.3.2 | Datasets | 84 |
| 3.3.3 | Proposed models | 86 |
| 3.3.4 | E-LSOA: An Efficient LSTM-based model | 87 |
| 3.3.5 | Experiments and results | 89 |
| 3.3.6 | Hate speech detection in Algerian dialect | 95 |
| 3.3.7 | Conclusion | 97 |
| 3.4 | Chapter conclusion | 97 |
| 4 | Arabic dialect identification | 99 |
| 4.1 | Introduction | 99 |
| 4.2 | Methodology | 101 |
| 4.2.1 | Dataset | 102 |
| 4.2.2 | Data Preparation | 103 |
| 4.3 | ML-based model for Arabic DI | 103 |
| 4.4 | Neural network-based models Arabic for DI | 105 |
| 4.4.1 | Model Architectures | 105 |
| 4.4.2 | Training and Evaluation | 105 |
| 4.5 | Bert-based models for Arabic DI | 106 |
| 4.6 | Experiments and results | 107 |
| 4.7 | Discussion | 108 |
| 4.8 | Conclusion and future work | 110 |
| 5 | DziriBERT | 112 |

| | | |
|----------|--|------------|
| 5.1 | Introduction | 112 |
| 5.2 | DziriBERT: an Algerian Language Model | 115 |
| 5.2.1 | Training Data | 115 |
| 5.2.2 | Language Modeling | 115 |
| 5.3 | Evaluation of DziriBERT | 116 |
| 5.3.1 | Twifi | 117 |
| 5.3.2 | Narabizi | 117 |
| 5.3.3 | Twifi | 118 |
| 5.4 | Discussion | 118 |
| 5.5 | DziriBERT for Named Entity Recognition | 120 |
| 5.6 | Limitations | 123 |
| 5.7 | Conclusion | 124 |
| 6 | Effective approaches for Arabic Text Classification | 125 |
| 6.1 | Introduction | 125 |
| 6.2 | Methodology | 129 |
| 6.2.1 | Neural-networks-based approaches | 129 |
| 6.2.2 | Proposed TransConvNet model | 129 |
| 6.3 | Experimental setup | 132 |
| 6.3.1 | Arabic text classification datasets | 132 |
| 6.3.2 | Hyperparameter tuning | 134 |
| 6.4 | Results and Discussion | 135 |
| 6.4.1 | Statistical evaluation | 138 |
| 6.4.2 | Error Analysis | 139 |
| 6.5 | Conclusion | 139 |
| | References | 145 |

List of Figures

| | | |
|------|---|-----|
| 2.1 | Number of NLP publications each year and per language (N. Habash 2019) | 14 |
| 2.2 | The Transformer model architecture (Vaswani et al. 2017a). | 22 |
| 2.3 | CBOW and Skip-gram representations. | 27 |
| 2.4 | Class distribution of the multi-class NADiA dataset. | 46 |
| 3.1 | Generic flow graph of the proposed system. | 53 |
| 3.2 | The proposed BiGRU additive-attention model | 58 |
| 3.3 | Classification report of our baselines compared to our model. | 63 |
| 3.4 | ASER Data collection pipeline. | 73 |
| 3.5 | Raw WAV file before and after pitch tuning. | 75 |
| 3.6 | Raw WAV file before and after adding white noise. | 75 |
| 3.7 | Generic graph for the proposed DeepEmoNet network for SER. | 77 |
| 3.8 | LSTM-CNN encoders for the proposed DeepEmoNet model. | 77 |
| 3.9 | Loss and Accuracy Analysis of the Proposed DeepEmoNet Model. | 79 |
| 3.10 | Proposed E-LSOA model architecture | 90 |
| 4.1 | Maghrebi dialects variations. | 101 |
| 4.2 | Generic graph showing the overall DI methodology for classical ML experiments | 104 |
| 4.3 | Generic graph for Maghreb DI using an LSTM model | 106 |
| 4.4 | Accuracy level according to the number of features used in TF-IDF without stopwords. | 109 |
| 5.1 | DziriBERT Tokenization process. | 116 |
| 6.1 | Overall architecture of neural-networks-based models experiments | 130 |
| 6.2 | Our TransConvNet model architecture | 131 |

| | |
|---|-----|
| 6.3 Pretrained language models experiments overall architecture | 132 |
| 6.4 Class distribution of the multi-class datasets. | 135 |
| 6.5 Headlines length (number of tokens per headline) distribution in the collected ANHD dataset. | 136 |

List of Tables

| | | |
|------|--|-----|
| 2.1 | Summary of related works on Arabic sentiment Analysis | 34 |
| 2.2 | SOTA in Arabic Speech Emotion Recognition | 38 |
| 2.3 | Summary of some available Arabic dialects datasets | 40 |
| 2.4 | Recent pretrained language models for the Arabic language (AbdelRahim El-madany, Nagoudi, et al. 2022) | 42 |
| 2.5 | Some recent works on Arabic document classification | 44 |
| 3.1 | The size of different datasets | 59 |
| 3.2 | Summary of hyperparameter (HP) selection (the bold item is the chosen HP) | 61 |
| 3.3 | Sentiment Analysis results for the HARD dataset | 62 |
| 3.4 | Sentiment analysis results for the BRAD dataset | 62 |
| 3.5 | Sentiment analysis results on LABR dataset | 62 |
| 3.6 | Comparison of sentiment analysis results on different datasets | 64 |
| 3.7 | Statistical test on the performance of the proposed model and the BiGRU model. | 64 |
| 3.8 | Overview of class distribution in the ASER dataset. | 74 |
| 3.9 | Experimental results on ASER dataset. | 78 |
| 3.10 | Hate speech datasets used in our study | 85 |
| 3.11 | Models' comparison across the datasets | 91 |
| 3.12 | Experimental results of different models across all datasets | 92 |
| 3.13 | Performance Comparison of Various Models for Algerian hate speech detection | 96 |
| 4.1 | Number of sentences per dialect in Maghreb-DI dataset | 102 |
| 4.2 | Samples of sentences written in Arabizi. | 103 |
| 4.3 | Samples of sentences written in Maghrebi dialect. | 103 |
| 4.4 | SVM and Logistic Regression model hyperparameters | 104 |

| | | |
|-----|--|-----|
| 4.5 | Default parameters and values for Multinomial Naive Bayes classifier in scikit-learn | 105 |
| 4.6 | Experimental results of ML models based on the number of features | 108 |
| 4.7 | Performance comparison of different models using Accuracy, F1-score, and Precision on Maghreb-DI | 109 |
| 5.1 | Accuracy and macro averaged Precision, Recall and F1 score obtained by each model on the Twifil sentiment dataset. | 118 |
| 5.2 | Accuracy and macro averaged Precision, Recall and F1 score obtained by each model on the Twifil emotion dataset. | 118 |
| 5.3 | Accuracy and macro averaged Precision, Recall and F1 score obtained by each model on the Narabizi sentiment dataset. | 119 |
| 5.4 | Accuracy and macro averaged Precision, Recall and F1 score obtained by each model on the Narabizi topic dataset. | 119 |
| 5.5 | Models comparison according to the vocabulary length, the total number of parameters and the final size on disk. | 121 |
| 5.6 | Metrics Comparison for Different Arabic BERT Models on DzNER (A. H. Dahou et al. 2023b) | 122 |
| 5.7 | Strict entity type-level F1-score for the four most frequent entity types for the three scripts NArabizi (NA), Alg-Arabic (Ar), and codeswitched (CS) in test (Touileb 2022) | 122 |
| 6.1 | The hyperparameters of our proposed TransConvNet model | 131 |
| 6.2 | Number of headlines per class in the ANHD dataset. | 133 |
| 6.3 | Number of headlines in each dataset for training, validation, and total. | 134 |
| 6.4 | Models' hyperparameters | 135 |
| 6.5 | Classification Results on the ANHD, SANAD-ALL and NADIA datasets | 136 |
| 6.6 | Classification results on 3 subsets of SANAD | 137 |
| 6.7 | Statistical test on the performance of the proposed TransConvNet model. | 139 |

Chapter 1

Introduction

Natural language processing (NLP) has witnessed significant advancements in recent years, with applications spanning multiple domains. In general, NLP involves the development of algorithms, models, and systems that can analyze, understand, and generate human-like language, with the ultimate goal of enabling computers to effectively communicate with humans in a useful and meaningful way (Manning et al. 1999). Deep neural networks have significantly contributed to the advancement of NLP research and applications, achieving impressive performance in various fields, such as conversational systems and virtual assistants for tasks such as customer service (Torfi et al. 2020). These technologies have also been applied to tasks such as sentiment analysis, topic classification, language detection, and translation, potentially saving time and increasing efficiency.

1.1 Context and Motivation

Despite the impressive performance of NLP systems across multiple applications, it is worth noting that not all languages have received equal representation in the NLP research community. This may be due to factors such as data availability, resource constraints for research in certain regions of the world, and the inherent complexity of particular languages, which can affect the distribution of research efforts across different languages. The assumption that all languages possess a sufficient degree of linguistic similarity to be processed using the same methods and tools is questionable. Therefore, it may be more practical to develop specific tools and models for each language. More specifically, the Arabic language has received relatively less attention

compared to other languages in terms of (i) the development of specific learning models and (ii) the availability of data resources that could help in advancing the field and opening new application opportunities in the market. This is particularly noteworthy given the economic and political significance of the Arabic language. Additionally, there is a lack of comparative studies and surveys on this topic. The purpose of this thesis is to address the mentioned concerns within the Arabic language research community and to bridge the gap between Arabic and other dominant languages in deep learning and NLP research:

1. Proposing new data resources: A significant portion of our research focused on collecting new data from online sources such as online websites, and social media platforms for various NLP tasks, such as dialect identification, hate speech detection, short-text classification, and pretraining various corpus data in either Arabic or dialect language.
2. Proposing new state-of-the-art (SOTA) models' architectures and language models: During our research, we focused on proposing new neural network model architectures that provide SOTA performance on various NLP tasks.

1.2 Research Contributions

The focus of our research is the exploration of numerous NLP applications to tackle some challenges imposed by the Arabic language and contribute to advancing the Arabic NLP research area. Our contributions include data collection, literature analysis, research publication in peer-reviewed venues, model development with training and/or pretraining of model parameters.

1.2.1 Arabic text and speech analysis

In modern society, individuals and consumers frequently express their opinions on a range of topics such as shopping experiences, cinematic content, religious rituals and conflicts, political and sports events, and more. This phenomenon has created significant concerns and challenges for governments and business owners who are required to analyze and consider analyzing these opinions and discussions in order to orient their decisions and actions. Our research in this regard aims to solve the following problems tailored for the Arabic language and its dialects:

The problem of inferring sentiments and emotions from textual and speech data

Sentiment analysis (SA) is a vital NLP application that aims to determine the sentiments expressed in a piece of text. Automating this process can be valuable for businesses entities seeking to gain insights into customer reviews of their products and services. Towards this end, we examine the challenges posed by the unique characteristics of the Arabic language and how our proposed techniques address such challenges. Furthermore, we propose a novel sentiment classifier based on attention mechanism and evaluate its performance on three large ASA benchmarks. Our proposed model outperforms previous related works and classical Neural networks-based models.

The problem with inappropriate content in social media

Despite the many benefits that the internet and social media platforms offer to their users, they have also been the source of significant concerns due to the offensive, hateful, and biased speech that is often conveyed and shared on these platforms. This type of content has the potential to cause harm to individuals and can contribute to a toxic online environment, which is why addressing these issues is of paramount importance. To address these pressing concerns, this thesis investigates the presence of hate speech, offensive content, and racial bias in Arabic text on social media platforms and news websites through the analysis of recent surveys. In addition, we put forward new deep learning models that are trained to detect various forms of inappropriate speech in MSA and dialectal Arabic datasets. As part of this effort, we also propose a newly collected Algerian dialect dataset for hate speech and offensive content collected from online social media platforms.

The problem with recognizing emotions in Algerian spoken dialect

In this section, we delve into the area of NLP related to speech analysis, specifically focusing on the detection of emotions conveyed through local Algerian spoken dialects. We recognize the importance of this research topic due to the unique linguistic and cultural characteristics of the Algerian dialect and its usage in various forms of media, such as TV shows. To achieve our goal of accurately detecting emotions in speech, we have taken a data-driven approach and collected a large speech dataset from Algerian TV shows. The dataset was then manually annotated to provide ground truth for the emotions conveyed in the speech. Building upon this dataset, we train multiple deep learning models to identify the emotions conveyed in the

speech. The models take advantage of the latest advancements in deep learning and speech analysis techniques and are designed to learn from the annotated data to accurately identify emotions in speech. The results of our models have shown promising results and provide valuable insights into the detection of emotions in speech in the local Algerian dialect.

1.2.2 Arabic dialect identification: Challenges and Approaches

The Arabic language is renowned for its diverse array of dialects and descendant languages, with an estimated 25 or more dialects spoken across various regions. From other perspectives, the Arabic language is a collection of a wide range of dialects spread in the Arabic peninsula and North Africa. This linguistic diversity presents challenges to the development of systems that can accurately capture and process all spoken dialects. Moreover, one important task in NLP is the identification of languages and dialects. This is crucial for applications such as machine translation, chatbots, and text-to-speech systems. In this thesis, to address this issue we propose a new dataset and multiple machine learning techniques, specifically focusing on the north-African dialects. We also explore different word embedding techniques and classifiers for identifying and processing Arabic dialects using shared benchmarks.

1.2.3 Developing DziriBERT: A Pretrained Language Model for Algerian Dialect

Language models have made significant contributions to NLP, but most of them are designed for English and other rich-resource languages. There is a need for a language model that can understand and generate the Algerian dialect, which has unique characteristics different from Arabic MSA and other dialects. To address this issue, we collected more than 3.3 million Algerian tweets and posts from online social media, and pre-trained the first Algerian language model: DziriBERT. When compared with other existing models, DziriBERT achieves better results, especially when dealing with the Roman script. The obtained results show that pre-training a dedicated model on a small dataset (450 MB) can outperform existing models that have been trained on much more data (hundreds of GB). Finally, our model is publicly available to the community.

1.2.4 Arabic document classification

The automatic classification of text documents is a critical and challenging task, particularly when dealing with very long or very short text sentences. In this thesis, we present a comparative study of 13 deep learning models for multi-class and multi-label classification of Arabic text data in both long and short scenarios. The models were evaluated using six Arabic benchmarks and two types of embedding models. We also collect and introduce the largest known dataset of short Arabic news headlines and propose a new Transformer architecture that outperforms other neural network models.

1.3 Statement of personal contribution

In each of the published papers included in the corp of the chapter of this thesis, I played a primary role in formulating the research questions, designing the methodology and experiments, preparing and collecting the data, implementing the models, analyzing the results, and drafting the initial versions of the papers. Except in the DziriBERT contribution (Abdaoui, Berrimi, et al. [2021](#)), where the first author contributed to the pretraining and evaluation of DziriBERT.

1.4 Role in Research: Challenges, Achievements, and Lessons

As a key actor in the research contributions listed in this thesis, my responsibilities included designing methodologies, collecting and analyzing data, developing and evaluating models, and contributing to the writing and revision of research papers.

During the course of the Ph.D. program at the University of Ferhat Abbas, I encountered several challenges, including but not limited to: (1) narrow availability of labeled data for Arabic text analysis across all NLP applications, (2) lack of hardware resources for both Cloud infrastructure access and GPU cards to conduct the experiments where in some cases I had to pay hundreds of dollars to evaluate models, and (3) technical issues in training and fine-tuning pre-trained dialect-specific models. To overcome these challenges, I employed various strategies such as leveraging crowdsourcing platforms for data collection, exploring different evaluation measures, enrolling in advanced courses and programs, attending international R&D summits, and collaborating with domain experts to address technical problems. I had the privilege to

become the first Algerian to be recognized as Machine Learning Expert by Google [\[1\]](#) in which I benefit from getting close to Experts, Developers and Googlers.

Despite these challenges, I also made several noteworthy achievements, including (1) developing novel deep learning systems to analyze Arabic speech, (2) improving the SOTA results in Arabic dialect identification, sentiment analysis, and Arabic documents classification, and (3) contributing to the development and release of DziriBERT: the first pre-trained language model for Algerian dialect understanding, which was presented at a workshop in a prestigious conference (NeurIPS). These achievements were possible due to my thorough understanding of the research problem, creative problem-solving skills, and strong technical expertise in NLP and deep learning.

Throughout the research project, I learned several valuable lessons, including the importance of (1) planning and organizing research tasks, (2) collaborating effectively with team members, (3) adapting to unexpected changes in the research plan, and (4) critically evaluating research results and reflecting on the limitations of the proposed methods. These lessons have not only helped me to successfully complete the research project but also shaped my approach to future research endeavors.

1.5 Thesis organization

The organization of this thesis is based on a list of our scientific publications. We start by presenting a brief overview of some characteristics of the Arabic language and the challenges it imposes from the linguistic side and present several SOTA deep learning models used in NLP applications, along with word representations and related works to our papers from the literature review. Moreover, we present our contributions in different chapters.

The content of the corresponding chapters is largely based on our published papers during the PhD thesis period, with minor modifications in some chapters, and removing the overlapping between some chapters. The order and content of the chapter's sections were occasionally adjusted for consistency, the related works for each paper were removed and consolidated in Chapter 2. These changes were made to ensure coherence within the thesis as a whole. It is important to note that we have also included new contributions to some chapters on certain

¹<https://developers.google.com/community/experts/directory/profile/profile-mohamed-berrimi>

points that were not covered in the published papers where more detailed analyses were provided when judged necessary.

This thesis is written in a way in which each chapter is self-contained and can be read independently of the others.

1.5.1 Chapter 2: Background, SOTA, and related works

Chapter 2 briefly outlines the specific characteristic of the Arabic text from a linguistic and computational perspective. Subsequently, we present a literature review of the SOTA deep learning models developed for processing textual data. The chapter concludes with a review of relevant research on the Arabic NLP on various domain applications.

1.5.2 Chapter 3: Arabic text and speech analysis

In the third chapter, we present our contributions to multiple areas of text and speech analysis. First, we present our research efforts toward solving Arabic sentiment analysis, in which we present an extensive literature study and develop new learning models and evaluate their performance on various sentiment analysis datasets. We furthermore provide a thorough analysis of various word embedding techniques. Secondly, the chapter presents an in-depth analysis of inappropriate/offensive speech in Arabic text using deep learning techniques. A range of deep learning models are trained and compared, and a new offensive speech dataset for the Algerian dialect is introduced and analyzed. Also, We present our work for Algerian speech emotion recognition, where a new audio dataset is collected and annotated, and multiple DL models are trained and evaluated.

The goal of this chapter is to provide insights into the effectiveness of deep learning approaches for the analysis of emotions from speech and text and also in detecting and preventing inappropriate speech in Arabic text. The material in this section builds upon three published papers: “Attention Mechanism Architecture for Arabic Sentiment Analysis” (Berrimi, Oussalah, et al. [2022](#)), “Attention-based Networks for analyzing inappropriate speech in Arabic Text,” (Berrimi, Abdelouaheb Moussaoui, Oussalah, et al. [2020b](#)) and “Effective speech emotion recognition using deep learning approaches for Algerian Dialect” (Yahia Cherif et al. [2021](#)).

1.5.3 Chapter 4: Arabic dialect identification

In this chapter, we provide a comprehensive overview of our study on Arabic dialect identification from textual data. Our study is motivated by the need for automated systems that can accurately identify different Arabic dialects, which pose unique challenges due to their rich linguistic and cultural diversity. To this end, we first describe our experimental setup and evaluate the performance of classical machine learning models on a newly collected dataset of text sentences from North African dialects. Afterward, we delve into more advanced cases and explore the use of recent language models to enhance the accuracy and robustness of more Arabic dialects. Our findings and insights are based on a systematic analysis of the results obtained from our experiments and are further supported by a thorough review of the related literature. The content of this chapter is largely inspired by our conference paper "Arabic Dialects Identification: North African Dialects Case Study" (Berrimi, Abdelouahab Moussaoui, et al. [2020](#)) and represents a major contribution to the ongoing research in this area.

1.5.4 Chapter 5: DziriBERT: a Pretrained language model for Algerian dialect

Chapter 5 details the development and pretraining of DziriBERT, the first language model that is specifically pretrained for the Algerian dialect. We describe the data collection and processing process, the architecture of the model, and the results of our experiments, in which we evaluate the performance of DziriBERT on multiple down-stream NLP tasks on Algerian text dialect such as sentiment Analysis, named entity recognition, emotion recognition and other. This work was presented in an affinity workshop at NeurIPS 2022. The goal of this chapter is to demonstrate the effectiveness of DziriBERT for NLP tasks in the Algerian dialect.

1.5.5 Chapter 6: Effective deep learning models for Arabic text classification

Chapter 6 presents a research methodology for improving the classification of Arabic online news documents using deep learning techniques. The chapter covers both long and short documents, as well as multi-class and multi-label settings. To address the lack of resources in this area, we introduce a new large short-text Arabic dataset and present a new transformer-based model. We also compare contextual and word embedding models and propose a new deep learning model that outperforms previous neural net models for all tasks. The goal of this chapter is

to demonstrate the effectiveness of deep learning approaches for classifying Arabic online news documents.

1.6 Publications

The papers mentioned below are a component of the research conducted during the Ph.D. course. Other papers were published, however, they are not relevant to the content of the thesis and have been excluded for the sake of coherence.

1.6.1 International Journals

1. Berrimi, M., Oussalah, M., Moussaoui, A., & Saidi, M. (2022). Attention Mechanism Architecture for Arabic Sentiment Analysis. *ACM Transactions on Asian and Low-Resource Language Information Processing*.
2. Berrimi, M., Oussalah, M., Moussaoui, A., & Saidi, M. (2022). A Comparative Study of Effective Approaches for Arabic Text Classification. (*under-review*).

1.6.2 International Conferences

1. Abdaoui, A., Berrimi, M., Oussalah, M., & Moussaoui, A. (2022). Dziribert: a pre-trained language model for the Algerian dialect. In the Affinity Workshop of North Africans in ML at **NeurIPS2022**.
2. Cherif, R. Y., Moussaoui, A., Frahta, N., & Berrimi, M. (2021). Effective speech emotion recognition using deep learning approaches for Algerian dialect. In 2021 International Conference of Women in Data Science at Taif University (WiDSTaif) (pp. 1-6). IEEE.
3. Berrimi, M., Moussaoui, A., Oussalah, M., & Saidi, M. (2020). Arabic dialects identification: North African dialects case study. *IAM'20: Third conference on informatics and applied mathematics, 21–22 October 2020, Guelma, ALGERIA.*;
4. Berrimi, M., Moussaoui, A., Oussalah, M., & Saidi, M. (2020). Attention-based networks for analyzing inappropriate speech in arabic text. In 2020 4th International Symposium on Informatics and its Applications (ISIA) (pp. 1-6). IEEE. 15-16 Decembre 2020, Msila, Algeria.

5. Berrimi, M., Moussaoui, A., Oussalah, M., & Saidi, M. (2020). Effective Deep models for Arabic sentiment analysis. In ICMCS'20: 7th International Conference on Multimedia Computing and Systems. 1-3 October, 2020. Morocco.

Chapter 2

Background

In this chapter, we provide a brief overview of the Arabic language from both linguistic and computing perspectives. We discuss the challenges and specific characteristics of the Arabic language that make it unique in NLP research and how these differ from those found in other languages. Our goal is to provide a comprehensive understanding of the complexities and nuances of the Arabic language and its impact on NLP research and development. After that, we review the SOTA deep learning models and different word representation techniques that have been developed for NLP tasks. We provide an explanation of these models and techniques, highlighting their strengths and limitations. Finally, we mention related works that have inspired our publications and contributed to the advancement of NLP research in the Arabic language. It is important to note that the techniques discussed in the first two sections of this chapter are only a selection of those that have been employed in our research application. There are many additional techniques that have been developed in the field of Arabic NLP that could potentially be useful for various tasks and applications.

2.1 Arabic language in NLP, background and challenges

The Arabic language is a Semitic language that originated in the Arabian Peninsula. It is spoken by over 400 million people worldwide, used by over 2 billion Muslims, and is one of the six official languages of the United Nations. Arabic is known for its rich ambiguity compared to languages like English and Latin, as it is highly inflected and has a complex system of nouns and verbs. This allows for a high degree of flexibility and expressiveness in the language (N. Habash

2019). One of the most distinctive features of Arabic is its script, which is written from right to left and contains a number of diacritical marks that are used to indicate vowel sounds and other features of the language. The Arabic script has been used to write many other languages over the centuries, including Persian and Urdu. The Arabic has three primary spoken forms:

- **Modern Standard Arabic (MSA):** is the standardized form of Arabic used in formal settings such as speeches, broadcasting, and education. MSA was developed in the Arab world in the late 19th and early 20th centuries. Most of the Arabic datasets existing today are expressed in MSA, due to the availability of online news websites that are broadcasting and publishing Arabic MSA content.
- **Classical Arabic:** is the oldest form of Arabic, used from the 7th century and throughout the Middle Ages. It is the language of the Quran, the holy book of Islam, and in Sunnah, the books of Prophet Mohamed's Hadith.
- **Dialectal Arabic:** These are inflected Arabic varieties that have been heavily influenced by foreign languages, such as English in the Middle East and French in North African countries, due to past colonization. The Arabs often prefer to use their own dialects in social media (Zaidan et al. 2014) using both Arabic and Latin scripts, and significantly different from one to another and from MSA in terms of phonology, morphology, lexical choice, and syntax. Geolinguistically, the Arab World can be divided into several regions, each with its own set of dialects (N. Y. Habash 2010). Egyptian Arabic is one of the most well-known dialects and covers the dialects of the Nile valley, including Egypt and Sudan. It gained widespread popularity due to the popularity of Egyptian cinematic content in the early 20th century. Levantine Arabic includes the dialects of Lebanon, Syria, Jordan, and Palestine, also known as "Al-sham." Gulf Arabic includes the dialects of Kuwait, United Arab Emirates, Bahrain, Omani, and Qatar, as well as Saudi Arabia (referred to as Hijazi also), which has a wide range of sub-dialects. This dialect is the closest to MSA. North African (Maghrebi) Arabic covers the dialects of Algeria, Morocco, Tunisia, and Mauritania. Libyan Arabic is sometimes included in this region, and these dialects are heavily influenced by French and Berber languages and have many sub-dialects in each country. Iraqi Arabic has elements of both Levantine and Gulf dialects, while Yemenite Arabic is often considered its own class.

More formally, in contrast to other big-five EU languages (English, Spanish, French, German),

Arabic is challenged by at least three key challenges. First, the Arabic language has a standard version that is well-understood across the Arab world, known as MSA. However, most social media content is rather associated with Dialectal Arabic (DA), which often substantially differs from the MSA, while DA is strongly hit by the lack of standards and scarcity of tools employed in the processing pipeline (Abdul-Mageed, C. Zhang, AbdelRahim Elmadany, et al. [2022](#); Meftouh et al. [2015](#)).

This negatively impacts the contextual understanding of the content, and, thereby, the efficiency of the performance of the NLP tools. Second, figurative linguistic is very rich in the Arabic language in both MSA and DA, as manifested by its various linguistic devices such as metaphors, analogy, irony, sarcasm, euphemism, hyperbole, context shift, false assertions, oxymorons/paradox, rhetorical questions, to communicate more complicated meanings (Farha et al. [2021a](#)). This issue is widely unexplored in the current SOTA of Arabic sentiment analysis tools. Third, as a result of the preceding, conveying a fine-tuned evaluation of Arabic sentiments often requires a high level understanding of the content, which may go beyond the boundary of the given short text message, requiring for instance the knowledge of prior texts and sometimes even the subsequent and following textual messages as well. This especially holds for polyseme words where word sense disambiguation requires discourse analysis to reveal the correct sense of the target word. The above makes the already available Arabic parsers highly complex for simple NLP tasks. For instance,

S. Khalifa et al. ([2020](#)) pointed out that a complete part-of-speech (POS) tagset in MSA has over 300K tags and would require 12.2 morphological analysis per word, compared to just 50 tags and 1.25 analysis per word for English. This high ambiguity is primarily due to Arabic orthography, which almost always omits the diacritics that are used to specify short vowels and consonantal doubling. Furthermore, the Arabic language has complex morpho-syntactic agreement rules and a lot of irregular forms. On the other hand, the lack of large-scale relevant benchmark datasets and ground truth restricts the development of efficient machine learning methods. This motivated some scholars to consider the Arabic language among low-resource languages on the semantic side because of the limitation of current parsers. For instance, the Arabic WordNet project (Jha et al. [2016](#)) contains less than 30% synsets of the English WordNet project. Similarly, given the rich morphology of Arabic language associated with the inherent challenges caused by Arabic dialects, one expects difficulties with the choice of an appropriate

pre-processing pipeline, including text segmentation, normalization, choice of stopword list, and stemming, among others. The Arabic language is spoken by a significant number of individuals

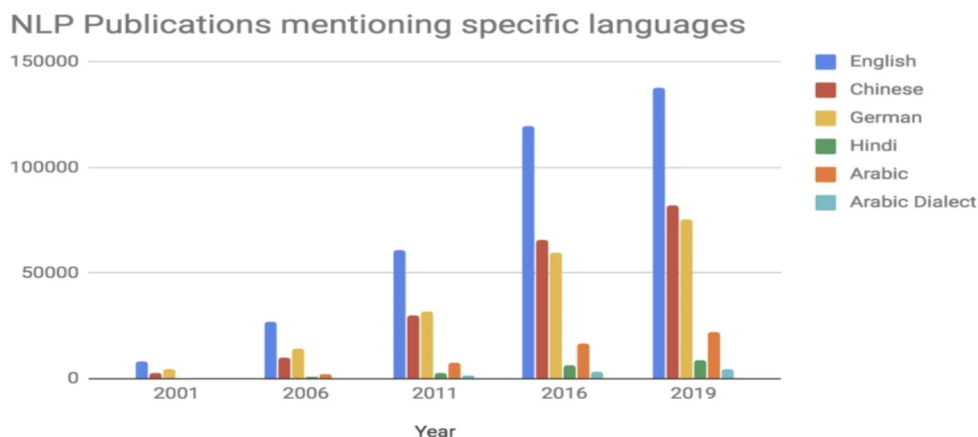


Figure 2.1: Number of NLP publications each year and per language (N. Habash [2019](#))

across a vast geographic region, including the Middle East, North Africa, and parts of Asia. It is the official language of several countries in this area, and it is also spoken by sizable communities in other regions such as Europe and North America. It is estimated that there are more than 400 million native speakers of Arabic and numerous others who use it as a second language. Given the widespread use of the Arabic language, there is a significant market opportunity for the development of automatic systems for processing and understanding Arabic text and speech, including chatbots, language models, automatic processing and analysis systems, and smart digital assistants. These systems have the potential to tap into a large and diverse market and can be applied to a variety of applications and industries, and ease the life for many people. In order to effectively serve the Arabic-speaking community, it is also important for companies and markets to invest in the development of systems that can accurately represent the Arabic language.

Despite the significant market opportunities presented by the Arabic language, research in this area has received relatively less attention in the past decade, as demonstrated in Figure [2.1](#).

There are several reasons for the under-representation of Arabic language research in NLP, including a lack of available data resources and a lower level of online social communication and content creation among Arabic speakers compared to other languages such as English, Mandarin, and French. Research funding and support for Arabic NLP research, The Morphological complexity imposed by the language. Additionally, a shortage of research contributions and

competition to develop effective machine learning models has hindered progress in this area.

Furthermore, the

However, there has been a notable effort to advance research in this field through the organization of meetings, workshops, affinity groups, and competitions, which are typically held annually and colocated at top machine learning venues and conferences such as NeurIPS, EMNLP, and ACL. In summary, Arabic is a rich and vibrant language with a long history and global presence that continues to adapt to the changing needs of its speakers. As such, it is an important language to study and understand.

2.2 Theoretical background: a systematic overview

NLP is a field that merges machine learning technologies and linguistic practices in order to address the interaction between human languages and computers. In other words, it is the field that uses machine learning techniques to allow computers to interpret, understand, and generate human-spoken languages.

The primary goal of constructing robust NLP pipelines is to learn from data in order to automate systems, make predictions, and facilitate real-world interactions (e.g. Conversational systems). The advancement of word representation techniques and deep learning models (feature extractors) is an active area of research and has seen significant progress in recent years. In this section, we provide a brief overview of the most commonly used machine and deep learning algorithms in the field of processing sequential and textual data as employed by the research and development community. Subsequently, we discuss various techniques for word embedding, which are crucial for representing the meaning of words in a numerical format that can be processed by these algorithms.

It should be noted that the chapter does not provide a detailed exploration of the technical and theoretical aspects of the models, as the primary focus of this thesis lies in highlighting our contributions. Comprehensive information on machine learning algorithms can be readily accessed through various sources.

2.2.1 Data engineering and quality assurance

Data plays a crucial role in the development of effective NLP systems. A significant amount of the data that is readily available online is in the form of text, which can be obtained from a range of sources including websites, social media platforms, and others. The availability of this training data is a decisive factor in the success of deep learning models (Sun et al. 2017).

To collect data, various methods can be employed, including web scraping, API access, and Crowdsourcing jobs. It is important to ensure that the collected data is of high quality and relevance to the task at hand in order to achieve optimal performance from the machine learning models. Data quality refers to the extent to which data is unbiased and representative of all real-world scenarios. Ensuring the quality of data is important in the development of AI systems, as bias or lack of representation in the training data can lead to negative consequences when the system is deployed. To avoid these risks, it is essential to use data that is free from bias towards any population, religion, or gender and that represents all relevant real-world scenarios equally. Another data source is the open-published data that is shared by researchers, organizations, governments, and industries for the purpose of result replication, competition, challenges, commercial use, and social good. This type of data is often made freely available and can be a valuable resource for machine learning research and development.

Despite the data abundance, most of it is unstructured. To ensure quality deep learning models, it is advised to feed in normalized, clean data in a way the learning model expects as an entry to learn from it and to perform good performance in the production stage. Over the years, the Arabic NLP research community has developed numerous specific methods for processing and cleaning Arabic text, as it requires explicit techniques that differ from those used for other languages. These techniques have been developed in order to effectively deal with the unique characteristics of Arabic text and enable the development of accurate and reliable NLP systems for this language.

Our contribution to the aforementioned matters is the collection of good quality and balanced data. For instance, in the domain of Hate Speech detection, we have collected the largest known balanced annotated dataset in the Algerian dialect. This dataset aims to facilitate the development of automated systems that can effectively combat various forms of inappropriate speech

on social media platforms. Additionally, in the areas of news headlines classification, speech emotion recognition and dialect identification, we have introduced novel balanced annotated datasets. These datasets are made accessible to the research community upon request, fostering further advancements in these fields.

2.2.2 Machine learning

ML is a branch of artificial intelligence that involves the development of algorithms and models that enable systems to learn from data and make predictions or decisions without being explicitly programmed. In the context of NLP, machine learning is a key enabler for the development of successful NLP solutions. This is because NLP problems often involve large amounts of unstructured text data, which is difficult to process using traditional rule-based methods. Machine learning approaches, such as deep learning and neural networks, have proven to be particularly effective at handling such data and have led to significant advancements in NLP tasks such as language translation, sentiment analysis, and text generation. We briefly mention the used machine learning algorithms in our contributions.

2.2.3 Logistic Regression

Logistic Regression is a statistical method for binary classification which is widely used in NLP for text classification tasks such as sentiment analysis, spam detection, and topic classification. It is a linear classifier that models the probability of a given input text belonging to a particular class. The model takes a set of handcrafted features, usually represented as a vector, and a set of weights and biases, and produces a probability that the input text belongs to a specific class. The predicted class is then obtained by comparing the predicted probability with a threshold value, typically 0.5.

2.2.4 Support Vector Machines

Support Vector Machines (SVMs) are a class of supervised learning algorithms that can be used for text classification. The basic idea behind SVMs is to find a linear boundary that separates different classes in a high-dimensional feature space. The boundary is chosen in such a way that it maximizes the margin, which is the distance between the boundary and the closest data points from each class. These closest data points are called support vectors and have a crucial role in determining the boundary.

2.3 Deep Learning

Deep learning is a subfield of machine learning that involves the use of neural networks with multiple layers, known as deep neural networks. These networks are designed to learn and represent the hierarchical representations of data. In the context of NLP, deep learning has been a key enabler for the development of successful NLP solutions. This is because NLP problems often involve large amounts of unstructured text data, which is difficult to process using traditional rule-based methods. Deep learning approaches, such as recurrent neural networks (RNNs) and transformer models, have been particularly effective at handling such data and have led to significant advancements in NLP tasks such as language translation, sentiment analysis, and text generation.

RNNs have been the de facto representation models for over 20 years for representing textual and sequential data. RNNs employ a feedback mechanism (also called recurrence) that allows information to flow from one step to the next enabling them to capture temporal dependencies in the data and to handle variable-length sequences. One of the main advantages is their ability to handle variable-length sequences, which is crucial for NLP tasks such as language modeling and text generation.

Despite its specific design for representing sequential data, RNNs come with major disadvantages which is the vanishing and exploding gradient problem (Goodfellow et al. [2016](#)). This occurs when the gradient of the error with respect to the parameters becomes very small or very high during the backpropagation process. This makes it difficult for the model to learn long-term dependencies in the data, as the gradients become too small / too large to update the parameters effectively.

To address these limitations, two main models have been put through, namely Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs).

2.3.1 LSTM

LSTM (Hochreiter et al. [1996](#)) are types of neural net models that are commonly used for processing textual data in NLP tasks. Unlike traditional RNNs, LSTMs have memory cells that can store information for a long time, allowing the network to handle long sequences of data, such as sentences or paragraphs. In NLP applications, LSTMs are often used for sentiment

analysis, machine translation, and text classification tasks. The basic idea behind an LSTM network is to process the input sequence one step at a time and keep updating the hidden state, which contains the information from previous steps. The hidden state is then used to make predictions about the next step in the sequence, such as the next word in a sentence or the sentiment of the text. The LSTM architecture has proven highly effective in many NLP tasks, especially when combined with other deep learning techniques, such as attention mechanisms and pre-trained language models.

2.3.2 GRU

GRU (Chung et al. 2014) is a revised version of LSTMs. Unlike LSTMs, GRU layers have only two gates, the reset gate and the update gate, which are used to control the flow of information in the network. The reset gate determines how much of the previous hidden state should be forgotten, while the update gate determines how much of the previous hidden state should be combined with the current input to compute the new hidden state. This allows the GRU to effectively capture long-term dependencies in the input data, and make the model train faster.

Bidirectional models

Bidirectional models are crucial for processing textual data because they can capture contextual information from both the forward and backward directions, resulting in a richer and more accurate representation of the data. Unidirectional LSTM models, on the other hand, process the data in a single direction, which can limit their ability to understand and represent context effectively, especially when the meaning of a word or phrase depends on the words that follow it. Bidirectional LSTM models are more powerful than their unidirectional counterparts because they leverage the information from both directions, enabling them to better capture complex patterns and dependencies within the text, ultimately leading to improved performance in various NLP tasks.

2.3.3 Convolutional neural networks

The main idea behind Convolutional neural networks (CNNs) for text processing is to apply the convolution operation, which is typically used for computer vision applications (LeCun et al. 1989), to the text data to extract local features from the text. In a typical CNN for text, the input text is first embedded into a dense vector representation, where each word is represented

as a dense vector in a high-dimensional space. Then, the embedded text is processed using a series of convolutional and pooling layers, which learn to detect patterns in the text and extract features that are relevant to the task at hand. Finally, the extracted features are passed through fully connected layers, which make the final prediction using the extracted features. We will introduce some CNN layers in further chapters in this thesis.

2.3.4 Attention Mechanism

The soft attention mechanism has emerged as a powerful approach for improving text classification tasks by enabling the model to focus on the most relevant parts of the input sequence. This mechanism assigns a weight to each token in the input sequence, and these weights are used to compute a context vector that captures the most important features of the input for the given task. In this section, we provide an overview of the soft attention mechanism and discuss its application in text classification.

Let us consider an input sequence $\mathbf{x} = (x_1, x_2, \dots, x_T)$, where x_t denotes the t -th token in the sequence and T represents the total number of tokens. The input sequence is first passed through an embedding layer to obtain a sequence of word embeddings $\mathbf{h} = (h_1, h_2, \dots, h_T)$, where h_t is the embedding for the t -th token. The embeddings can be further processed by additional layers, such as LSTM or CNN layers, to capture more complex features and dependencies.

In the soft attention mechanism, the model computes an attention weight α_t for each token in the sequence. The attention weight is calculated using an attention function $a(\cdot)$, which typically depends on the token's hidden representation h_t and some additional context information. A common choice for the attention function is the dot product followed by a softmax activation:

$$\alpha_t = \frac{\exp(a(h_t, c))}{\sum_{j=1}^T \exp(a(h_j, c))}, \quad (2.1)$$

where c represents the context information, which could be a fixed vector or the output of another layer in the model. The attention weights can be interpreted as the probability distribution over the input tokens, with higher values indicating more relevant tokens for the given task.

Once the attention weights are computed, the model calculates a context vector \mathbf{v} as a weighted sum of the hidden representations:

$$\mathbf{v} = \sum_{t=1}^T \alpha_t h_t. \quad (2.2)$$

The context vector \mathbf{v} captures the most important features of the input sequence as determined by the attention mechanism. This vector can then be used as input to a classifier, such as a fully connected layer followed by a softmax activation, to predict the target class.

In summary, the soft attention mechanism provides an effective way to focus on the most relevant parts of the input sequence for text classification tasks. By assigning attention weights to individual tokens and computing a context vector based on these weights, the model can better capture the important features and improve classification performance.

2.3.5 Transformer model architecture

A revolutionized model that took place recently to overcome the problem presented in RNN-based architectures is the Transformer network published by (Vaswani et al. [2017a](#)). The paper introduced a new learning approach that drops all recurrence operations in deep learning models and replaces them with Self-attention layers to make it easier to train the model and to give them the ability to attend more (assign more weights) to tokens that contribute more to the context. The Transformer network has led to a breakthrough in several language models like BERT (Bidirectional Encoder Representations from Transformers): a Bidirectional model that is built with several blocks of the Transformer network (Devlin et al. [2018](#)).

Encoder section of the Transformer model The original Transformer model was introduced for Machine translation applications and is composed of an encoder and decoder blocks. The encoder part of the transformer architecture is composed of 4 main building blocks:

- **Positional encoder:** Generally speaking, in natural language, the order of the words and their position in the sentence are important to capture the context, and reordering the words will change the meaning of the sentence. The positional encoder layer is built exactly to maintain the order of the tokens in the input sequence all along the feedforward pass in the model.
- **Multi-head attention** The transformer architecture introduced the self-attention mechanism as a replacement for recurrent and convolutional layers in its building blocks. The

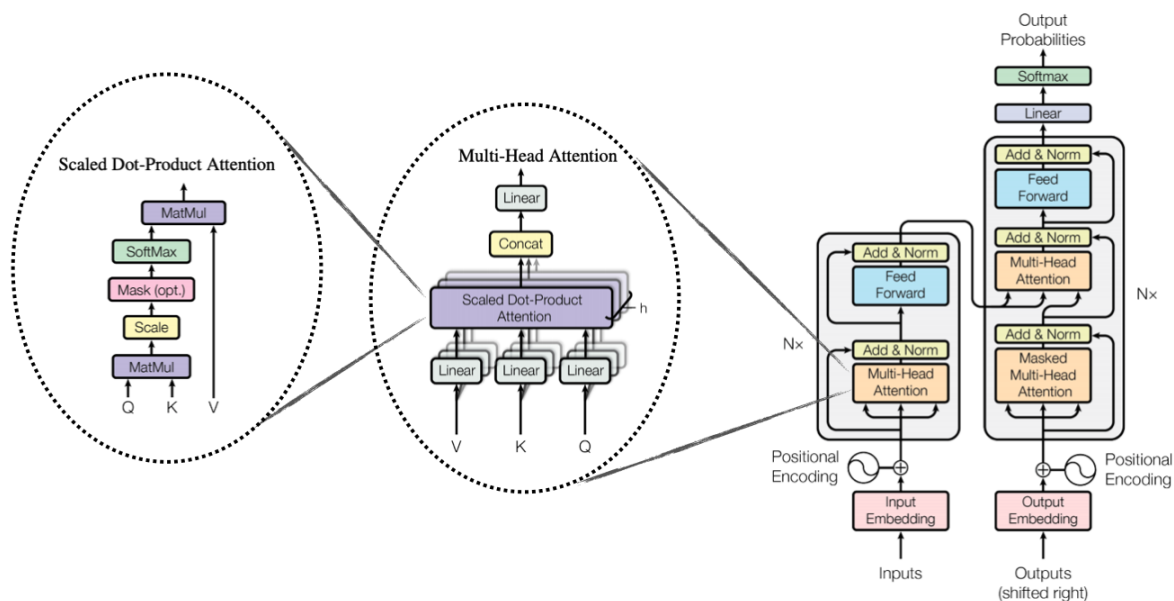


Figure 2.2: The Transformer model architecture (Vaswani et al. 2017a).

self-attention mechanism is mainly employed to help the model pay attention to other tokens while processing a token at time t .

In previously proposed attention layers (Zichao Yang et al. 2016; Bahdanau et al. 2014b) we look at parts of the input sequence and part of the output sequence, but in Self Attention, we look into input items and other input items within a sequence, in the transformer we do this several times, so each head will learn attention relationships independently.

The multi-head attention (MHA) layer in a transformer model processes input sequences by computing the attention between every position in the sequence and all the other positions. This is done by feeding the MHA layer with vectors that contain sub-vectors that embed the words in the input sequence. The MHA layer then uses these embeddings to compute the attention between each pair of words.

To compute the attention between two words, the MHA layer first treats one of the words as a query and uses the other words as keys. It then takes a convex combination of the values (which are the same as the keys) and computes a dot product of the keys and query divided by the square root of the normalized scaled size of the sequence. This process produces better embeddings of the input sequence by merging together information from pairs of words.

The MHA layer has multiple projections of the keys, queries, and values (KQV), which

corresponds to the number of heads, as illustrated in Fig 2-B. These projections are learned by learning weight matrices that calculate the weights of the current words and how much attention should be paid to other words in the sentence, as well as the vector representations of all the words in the sequence.

$$MultiHead(Q, K, V) = concat(head_1, \dots, head_n)W^O \quad (2.3)$$

$$where : head_1 = Attention(QW_i^q, KW_i^k, VW_i^v) \quad (2.4)$$

The multi-head layers feed the Fully connected layer FCL a weighted combination of words rather than one word at a time, so it passes the amount of each word it's going to include (if we are looking at a word at position n, which of the other words in the input sequence are relevant to that exact word).

The *add&Norm* layer takes the original input and adds it in a residual connection to the output of the MHA layer, then normalize them to have 0 mean and variance of 1.

The Encoder section of the transformer model is repeated N times, eventually, the multi-head will take every word in the input sentence and combine it with some other words through the attention mechanism to produce better embeddings that merge together information from pairs of words. A single pass of the encoder block will essentially generate embeddings for a pair of words, by repeating the encoder block N time, it will generate embedding for N pairs of words.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right) \quad (2.5)$$

The output vector (X) of the multi-head attention in the encoder section of the transformer is the weighted sum of the input vectors, where the weights are determined by the attention mechanism. This output vector is then used as input to the feed-forward neural network in the encoder.

Decoder section Along with the building components that are built in the encoder section, the decoder (the responsible for attending to the encoder input and generating the text) has a

masked self-attention mechanism that can be presented as follows:

$$\text{Masked Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T + \text{mask}}{\sqrt{d_k}}\right)V \quad (2.6)$$

d_k is the dimensionality of the keys, and "mask" is a tensor with elements set to negative infinity for the attention weights associated with future tokens in the target sequence. The "mask" tensor ensures that the attention mechanism only attends to the current and past tokens in the target sequence, as desired. By incorporating this masked self-attention mechanism, the decoder in the Transformer model is able to prevent itself from peeking into the future when generating the target sequence, which improves the model's ability to learn dependencies between elements in the target sequence and results in more accurate predictions.

2.4 Word representations

The word representation section constitutes a critical aspect of the architecture of a deep learning system. This section involves encoding the input text in a manner that facilitates the similarity in meaning to be reflected in the similarity of the vector representations. In this section, we provide a brief overview of several techniques that have been widely employed across various contributions and research papers in our experimental studies. It should be noted that the definitions presented here are contextualized based on their usage in the specific domain application.

2.4.1 Handcrafted features

Handcrafted word representation features in NLP are manually created features that describe the properties of words in a text. These features can include semantic information, such as the word's part of speech, root form, and sense, as well as syntactic information, such as the word's position in a sentence and its neighboring words. These features are used to represent words as vectors, which can be used as input to machine learning models for NLP tasks such as classification, tagging, and generation.

N-grams features

n-grams: An n-gram is a contiguous sequence of n items from a given sample of text, such as words, characters, or even letters. N-grams are commonly used as features in NLP tasks such as text classification, language modeling, and information retrieval. For example, in the sentence "the quick brown fox jumps over the lazy dog," a bi-gram would be a sequence of two words, such as "the quick", "quick brown", "brown fox", etc.

TF-IDF features

TF-IDF (Term Frequency-Inverse Document Frequency): This is a weighting scheme used to reflect the importance of a word in a document compared to a collection of documents. The term frequency (TF) measures the number of times a word appears in a document, while the inverse document frequency (IDF) measures the rarity of a word across the entire collection of documents. The product of these two values gives the TF-IDF weight, which reflects the importance of a word in a particular document compared to the collection as a whole. TF-IDF is often used as a feature in text classification, information retrieval, and document clustering tasks. The equation for the Term Frequency-Inverse Document Frequency (TF-IDF) can be represented as follows:

$$\text{tf-idf}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D) \quad (2.7)$$

where:

- t is the term (word) of interest
- d is the document that contains the term
- D is the collection of documents
- $\text{tf}(t, d)$ is the term frequency, representing the number of times term t appears in document d
- $\text{idf}(t, D)$ is the inverse document frequency, which can be calculated using the following formula:

$$\text{idf}(t, D) = \log \frac{N}{|\{d \in D : t \in d\}|} \quad (2.8)$$

where:

- N is the total number of documents in the collection D
- $|\{d \in D : t \in d\}|$ is the number of documents in which the term t appears at least once

2.4.2 Word2vec

Word embeddings are a type of dense, continuous-valued representation of words in a vector space. These representations are learned from large corpora of text data through unsupervised techniques, and are designed to capture the semantic and syntactic relationships between words in a given corpus. Unlike traditional bag-of-words representations, word embeddings encode rich linguistic information (context) in a compact and dense form, capturing semantic similarity and relatedness between words. They have become indispensable tools in modern NLP and have gained widespread adoption in the field due to their ability to capture the meaning and relationships between words in a computationally efficient and effective manner.

Word2vec (Tomas Mikolov, K. Chen, et al. [2013](#)) is a widely used unsupervised learning algorithm for learning word embeddings. Since its introduction, it has become a popular method for generating dense, continuous-valued representations of words in a vector space. Word2vec uses a neural network architecture to learn word representations from large amounts of text data. The network is trained on a prediction task, where it must predict the surrounding words of a target word given its context. The resulting word representations capture semantic and syntactic relationships between words, and have been shown to be effective in a variety of NLP tasks. Word2vec has two main variants: CBOW (Continuous Bag of Words) and Skip-gram. CBOW predicts the target word given its surrounding words, while Skip-gram predicts the surrounding words given the target word. Both variants have their strengths and weaknesses and can be used depending on the specific task and dataset.

Despite its superior performance on previous word representation techniques, word2vec embedding has some limitations. Word2vec relies on context-based prediction to capture relationships between words, which can lead to suboptimal representations for rare and/or unseen words (the

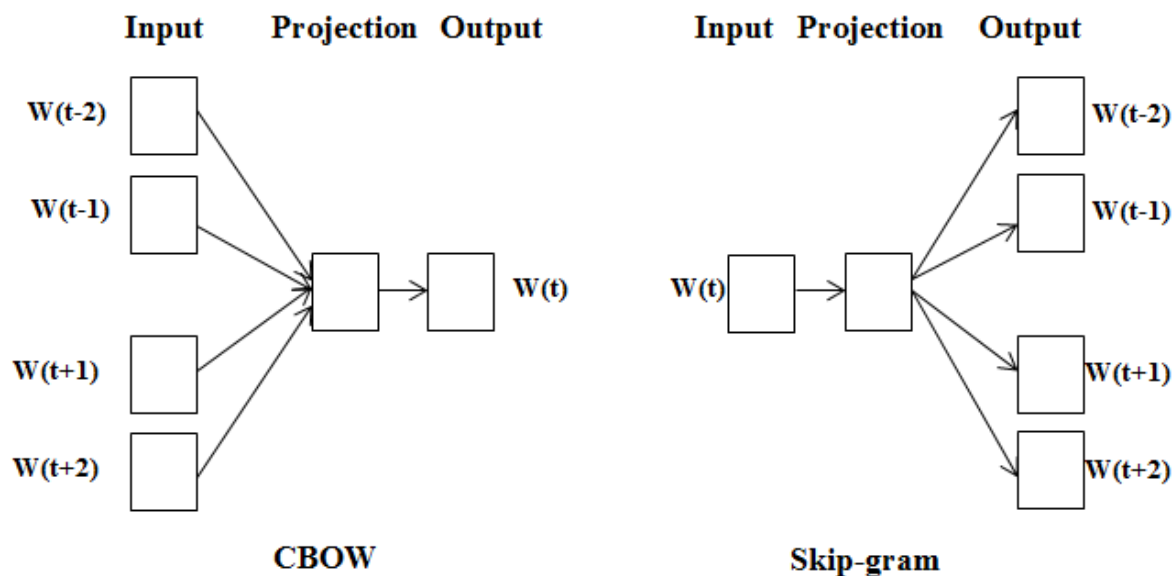


Figure 2.3: CBOW and Skip-gram representations.

problem with our of vocabulary). Additionally, it uses a simple dot product to measure the similarity between words, which can lead to the loss of important information conveyed in a given sentence.

2.4.3 GloVe

Global Vectors for Word Representation (GloVe) (Pennington et al. [2014](#)) is a widely used unsupervised learning algorithm for generating word embeddings. It was introduced as an alternative to the popular Word2vec method and aims to address some of its limitations. Unlike Word2vec, GloVe explicitly models the co-occurrence statistics between words in the corpus and provides a more interpretable measure of similarity between words. The algorithm is based on a factorization of the word-word co-occurrence matrix, which is calculated from a large corpus of text data. The factorization is optimized to minimize the difference between the observed co-occurrence statistics and the dot product of the word representations. This results in dense, continuous-valued representations of words that capture the semantic and syntactic relationships between words.

GloVe has been shown to provide improved performance on a variety of NLP tasks compared to Word2vec, particularly for rare words that have limited context. It has also been used in combination with other methods to improve the performance of NLP models and is widely adopted in the field of NLP due to its ability to effectively capture the meaning and relationships

between words.

One limitation of GloVe is its reliance on a fixed vocabulary of words, which can lead to sub-optimal representations for out-of-vocabulary (OOV) words and named entities, and is unable to capture relationships between sub-word units, such as morphemes and character n-grams. Another limitation of GloVe is its inability to handle polysemy, the phenomenon where a single word can have multiple meanings in different contexts.

2.4.4 FastText embeddings

The limitations of the Word2vec and GloVe models led to the development of FastText (Tomas Mikolov, Grave, et al. [2018](#)), a method that is specifically designed to handle out-of-vocabulary OOV words and polysemy by modeling sub-word units, such as character n-grams, in addition to words. FastText uses a neural network architecture to learn embeddings for sub-word units, which can then be combined to represent words. This allows FastText to handle OOV words and polysemy by leveraging the relationships between sub-word units.

Despite significant efforts to develop effective word representation models, the challenge of encoding words in a context-sensitive manner persists. This is due to the fact that words often have different meanings depending on the context in which they are used. In existing models, each word in the vocabulary is assigned a fixed input vector, which does not adapt to changing contexts. This limitation raises significant questions about the efficacy of these models.

2.4.5 Contextual embeddings and Transfer learning

Transfer learning is a technique that allows us to leverage the knowledge gained by a model during pre-training on a large, general-purpose dataset and apply it to a downstream task with similar data distribution. This approach has been shown to be effective in a wide range of NLP tasks, including sentiment analysis (Farha et al. [2021b](#)), text classification (Alammary [2022](#); Berrimi, Oussalah, et al. [2023](#)), and named entity recognition (Brandesen et al. [2022](#)). Some of the most commonly used pretrained models for Arabic are AraBERT (Antoun et al. [2020](#); Abdul-Mageed, AbdelRahim Elmadany, et al. [2021](#); Abdaoui, Berrimi, et al. [2021](#)). These models have been trained on large corpora of Arabic text and can be fine-tuned on specific tasks to improve their performance. In the following section, we discuss the building blocks of these language models.

BERT BERT (Devlin et al. 2018) is a SOTA NLP model that was introduced by Google. It is based on a transformer architecture (Vaswani et al. 2017a) and consists of 12 layers with 110 million parameters. BERT was pretrained on a large corpus of text in 104 languages, including Arabic, using two unsupervised learning tasks: masked language modeling (MLM) and next sentence prediction (NSP).¹

Since its release, BERT has achieved strong results on a variety of NLP benchmarks and challenges and has become a widely used model in research and industry. In particular, it has been shown to outperform many previous models on a variety of tasks, including sentiment analysis, question answering, and natural language generation. As a result, BERT has become a popular choice for researchers and practitioners working in NLP and has led to significant advances in the field.

Pretraining of BERT BERT uses a novel pretraining approach that involves Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) tasks. BERT's pretraining process involves unsupervised learning on a large corpus of text data. The pretraining process consists of two stages: pretraining and fine-tuning. During pretraining, BERT is trained on a large corpus of text data to learn contextual representations of words in the text. These contextual representations capture the meaning of words based on their surrounding context in the text.

The pretraining process of BERT involves two primary tasks: MLM and NSP. MLM is a task that involves masking some tokens in the input text and asking the model to predict the masked tokens based on the surrounding context. The objective of MLM is to force the model to learn contextual representations of words that can accurately predict the masked tokens in the text.

NSP is another task that involves predicting whether two sentences in the input text are consecutive or not. The objective of NSP is to teach the model to understand the relationship between two sentences in the text and to capture the flow of meaning between them.

During pretraining, BERT is trained using a large corpus of text data, such as Wikipedia or the BookCorpus. The text data is first tokenized into a sequence of tokens using WordPiece embeddings. The MLM task involves randomly masking some of the tokens in the input text and asking the model to predict the masked tokens. The NSP task involves randomly selecting

¹A list of pretrained language models (PLM) will be mentioned in related works of PLM in a later section 2.5.5.

two sentences from the text and asking the model to predict whether they are consecutive or not.

Fine-tuning is the process of using a pretrained model on a global objective (language modeling) to be trained on downstream tasks (i.e. sentiment analysis) on a smaller dataset, allowing the model to learn task-specific patterns and adapt its weights. This is typically achieved by adding a task-specific output layer, such as a classification or regression layer, on top of the pretrained model, and then optimizing the entire model using backpropagation.

Fine-tuning a BERT-based model involves several key steps. Firstly, a pretrained model needs to be selected as the starting point, which can be the original BERT model or one of its variants. It is important to choose a suitable language model that aligns with the requirements of the target task, as different language models may perform better on specific tasks based on their pretraining data. Evaluating the performance of various models on benchmark datasets can aid in making an informed choice (AbdelRahim Elmadany, Nagoudi, et al. [2022](#)).

To adapt the pretrained model for the target task, a task-specific output layer needs to be added. This modification involves adjusting the architecture of the pretrained model to align with the requirements of the specific task. For example, a dense layer with softmax activation can be added for classification tasks, while a linear output layer can be used for regression tasks.

Once the model architecture is modified, the pretrained weights of the BERT-based model are loaded, and the added output layer is initialized with random weights. This initialization allows the model to leverage the knowledge learned during pretraining while being able to specialize for the target task.

The next crucial step is fine-tuning the model. This involves training the entire model, including the pretrained layers and the task-specific output layer, using the target dataset. During fine-tuning, a smaller learning rate is often employed to ensure the model doesn't deviate too far from the pretrained weights. Fine-tuning BERT-based models offer several advantages over training models from scratch, such as faster convergence, and improved generalization. By transferring the pretrained knowledge, the model can capitalize on the vast amount of linguistic knowledge embedded within the model's weights, leading to superior performance on various NLP tasks.

2.5 Related works

In this section, the relevant related works from the literature review are presented in relation to our contributions. The related works of each chapter is presented in a separate section.

2.5.1 Related works in Arabic sentiment analysis

Traditionally, research studies targeting Arabic sentiment analysis use simple supervised models with conventional feature extraction techniques. The latter include Bag-of-Words (BOW), Term-Frequency-Inverse-Document Frequency (TFIDF), and N-grams (Gamal et al. 2019; Al-towayan et al. 2016). The main drawback of these models is that their performances decrease when the data became large and complex. While the features do neither encode nor represent the semantic relationships between tokens presented within the corpus. A comparative study on Arabic sentiment analysis (ASA) was conducted by (Farha et al. 2021a) where the authors replicated some recent SOTA methods on Arabic sentiment Analysis and examined the effectiveness of using Transformer based models, especially BERT models, on Arabic SA tasks. The fine-tuned models reported a better classification accuracy over three Arabic SA dataset: ASTD (Nabil et al. 2015) SemEVAL17 (Rosenthal et al. 2017) and ArSAS (AbdelRahim Elmadany, Hamdy Mubarak, et al. 2018). (Heikal Maha 2018) trained different neural network models such as LSTM, CNN, and RCNN on a collected dataset of 40K (positive/negative) labeled sentences. Their LSTM model achieved 81.3% accuracy. The authors then applied data augmentation to increase the size of the vocabulary, which enhanced the accuracy by +8.3%. (Rosenthal et al. 2017) in SemEval 2017 hosted a shared Arabic SA task where (Samhaa R. El-Beltagy et al. 2017) ranked first. Their system used a set of hand-engineered and lexicon-based features and a Naive Bayes classifier for training. (AbdelRahim Elmadany, Hamdy Mubarak, et al. 2018) presented the first Arabic sentiment analysis online system that accommodates both MSA and Arabic dialects. Their model was composed of a CNN layer followed by a pooling and LSTM block and was trained on the SemEval2017 dataset (Rosenthal et al. 2017). The model achieved 62% accuracy score. While their model achieved accuracy scores of 66% and 92% on ASTD (Nabil et al. 2015) and ArSAS (AbdelRahim Elmadany, Hamdy Mubarak, et al. 2018) datasets, respectively.

It is worth pointing out that word embeddings were used in most of the recent Arabic SA studies. For instance, (A. Dahou et al. 2019) built a neural word embeddings using Word2vec

(Tomas Mikolov, K. Chen, et al. [2013]) based on CBOW and Skip-Gram architectures on a vocabulary composed of 3.4 billion tokens from a 10 billion crawled corpus. The authors then trained a CNN model on the top of these embeddings and evaluated their quality on five Arabic sentiment analysis datasets, which, are then found to outperform 4 out of 5 previous works. (Ombabi et al. [2020]) used the FastText embeddings (Tomas Mikolov, Grave, et al. [2018]) skip-gram model trained using a CNN-LSTM based model. The extracted features are then passed to SVM classifier to generate the final classification. The CNN layer is inspired by the work presented by (Y. Kim [2014]) for local feature extraction followed by two LSTM layers to represent long-term dependencies. The model is then validated on two datasets (Nabil et al. [2015]) and LABR (Aly et al. [2013]) achieving 89.72%, 90.20%, and 88.52% in terms of precision, recall, and F1-measure, respectively. Owing to the emergence of several annotated dataset using dialectal Arabic, sentiment analysis of DA has seen a renewal interest (Hossain et al. [2019]; Moudjari and Akli-Astouati [2020]; Nabil et al. [2015])

For instance, Abu Kwaik et al. [2019] trained a BiLSTM-CNN model to analyze different Arabic dialects sentiments. The model reported a better result compared to two baselines that use LSTM and CNN models. In another research, (Soumeur et al. [2018]) studied sentiment analysis of the Algerian dialect. They worked on a collected dataset containing 100.000 comments, then labeled 25K reviews as negative, neutral, and positive. In their study, a CNN-based model achieved 89.5% accuracy. Authors in (Moudjari and Akli-Astouati [2020]) collected two Algerian dialects corpora from Facebook and Twitter and a combined Facebook/Twitter dataset. Next, the authors trained several ternary classification models, where the CNN-based model reported a better performance across all datasets.

Al-Dabet and Tedmori [2019] used an SRU layer (Simple Recurrent Unit) followed by an attention layer to weight the important terms. Their model achieved an accuracy of 94.53% in Arabic sentiment analysis task. We shall also mention the KAUST-sponsored competition on Arabic Twitter sentiment analysis ² (Alamro et al. [2021]). The top-ranked team used the AraBERT model by (Antoun et al. [2020]) and achieved an accuracy score of 84.5%. Besides, all the top three best-performing participating teams utilized the AraBERT model to generate the tweet

²<https://www.kaggle.com/c/arabic-sentiment-analysis-2021-kaust>

embedding representation. The second team combined static character and word-level models as suggested in (Abdullah et al. 2021). The training and testing used the ASDA (Abicra SE ANALYSIS) dataset (Basma et al. 2021), which is a large collection of 100K Arabic tweets, annotated for sentiment analysis tasks.

Recently, the sixth Workshop on Arabic Natural Language Processing (WANLP'2021) organized a task on sentiment detection (Ibrahim Abu Farha et al. 2021) on DAICT dataset (Abbes et al. 2020), which contains 5,358 tweets. The top performing team utilized MARBERT model (Accuracy score of 71.1%), while the second and third winning teams used AraBERT in combination with other deep learning models, achieving 70.4% and 69.5% accuracy score, respectively. Loosely speaking, the recent advancement in pretraining large Arabic language models has boosted the performance of many models in Arabic sentiment analysis. Fine-tuning the latest Arabic BERT-based model MARBERT that was pretrained on 6B Arabic tweets (Abdul-Mageed, AbdelRahim Elmadany, et al. 2021) has reported a better sentiment analysis performance on several Arabic SA datasets than the previous AraBERT model (Antoun et al. 2020). Also, our recent dialectal level DziriBERT model (Abdaoui, Berrimi, et al. 2021), which is pretrained on large Algerian tweets corpus, outperformed previous neural nets models and Arabic PLMs on various Algerian dialectal datasets.

Table 2 summarizes some of the key research in this field, highlighting the employed architecture, the dataset, the number of classes, and the accuracy score achieved by each model.

As can be observed from the above table, deep learning models tend to give better performances in terms of classification accuracy across many Arabic sentiment analysis datasets than standard ML-based classifiers such as LR and Linear SVC. This empirically supports the claim that deep-learning approaches become SOTA in the SA field. Therefore, our direction in this research paper is to employ and fine-tune several deep learning models for better classification performance in Arabic sentiment analysis.

On the other hand, the emergence of deep learning approaches has been boosted by the existence of open-source neural network libraries such as Keras, which attracted more researchers to work in this area. Nevertheless, it should also be emphasized that most deep learning approaches

Table 2.1: Summary of related works on Arabic sentiment Analysis

| Reference | Architecture | Dataset | Acc(%) | # class |
|--|-------------------------------|--|--------|---------|
| Alayba et al. (2018) | LSTM | 40K arabic tweets | 65.05 | 3 |
| Samhaa R. El-Beltagy et al. (2017) | NB | SemEVAL17 (Task A) | 58.1 | 3 |
| Abu Farha et al. (2019) | CNN-LSTM | SemEval2017 | 62 | 3 |
| Altowayan et al. (2016) | CNN (Word2vec+LR) | ASTD | 66 | 3 |
| Ombabi et al. (2020) | CNN-LSTM, FastText(Skip-gram) | ArSAS | 92 | 4 |
| Abu Kwaik et al. (2019) | BiLSTM-CNN | LABR | 88 | 2 |
| Soumeur et al. (2018) | CNN | ASTD, LABR | 80.21 | 2 |
| Moudjari and Akli-Astouati (2020) | CNN | TwitterASA cite{tSA} | 90.75 | 2 |
| Al-Dabet and Tedmori (2019) | Attention + Recurrent Units | LABR | 66.42 | 3 |
| Alamro et al. (2021) | MARBERT + CNN | LABR Balanced (binary) | 81.14 | 2 |
| Ibrahim Abu Farha et al. (2021) | MARBERT | LABR UnBalanced (binary) | 80.2 | 2 |
| Abdul-Mageed, AbdelRahim Elmadany, et al. (2021) | MARBERT & ARBERT | ASTD | 85.58 | 3 |
| | | Shami-Senti (binary) | 93.5 | 3 |
| | | A collected dataset | 92.03 | 3 |
| | | 25k annotated tweets (Mataoui et al. 2016) | 79 | 2 |
| | | LABR (binary) | 95.1 | 2 |
| | | ASAD dataset | 84.9 | 3 |
| | | Arabic tweets from (Abbes et al. 2020) | 71.1 | 3 |
| | | LABR | 92.51 | 2 |
| | | HARD | 96.17 | 2 |
| | | SemEVAL | 71 | 3 |
| | | ASTD-B | 96.24 | 2 |

for Arabic sentiment analysis suffer from the limited availability of large-scale Arabic sentiment annotated corpora for learning accurate models. Likewise, we also acknowledge the limited size of Arabic sentiment Treebank compared to English, as well as the increased algorithmic complexity due to a substantial increase of modalities embedded in the parser. This raises the importance to build on existing benchmark datasets to ease comparison and identify room for improvement.

In parallel to deep-learning approaches, one shall also mention the progress in metaheuristic approaches, including genetic algorithms, as in (Abualigah 2018) who suggested an efficient metaheuristic algorithm for feature selection employing multi-objective hybrid Krill-Herd algorithm for document clustering.

In the area of data benchmarking, one notices the emergence of some useful relatively good size resources that can improve distributional models. For instance, (Alayba et al. 2018) put forward a 1.5-billion words corpus, using ten newspapers from different Arab countries with different Arabic dialects to generate a distributed representation that has been used for sentiment classification purposes.

2.5.2 Related works in Arabic offensive speech detection

This section presents different aspects of hate speech, as well as research associated with the Arabic language. Due to the sensitivity of this matter, many studies were conducted to protect personals on social media, especially adolescents.

A. Elmadany et al. [2020] proposed a deep learning system based on Bidirectional Transformers BERT for offensive language detection. In their experiments, they used two types of data: data distributed by the Offensive Language Detection shared task and an automatically collected dataset.

The first data contains 10,000 tweets manually annotated for two sub-task: offensive speech, and hate speech. Their system's performance came up with 89.60% accuracy (82.31% macro F1) for hate speech and 95.20% accuracy and 70.51% macro F1 on official TEST data.

Alshehri et al. [2020] worked on Understanding and Detecting Dangerous Speech in Social Media on Arabic texts. They manually curate a multi-dialectal lexicon of physical harm threats. They used to collect a large dataset of threatening speech from Arabic Twitter, and manually annotate a subset of the data for dangerous speech, then trained BERT model for detecting hate speech. Their system yield a F1-score between 53.42% and 59.60% on detecting hate speech.

B. Haddad et al. [2020] proposed an Attention-based Deep Neural Networks to detect offensive speech in the Arabic language. They worked on the OffenseEval 2020 dataset where they conducted many experiments using the Bidirectional GRU model augmented with an attention layer, achieved a 85.9% F1 score for the task of offensive language detection, and a 75% F1 score for the task of hate speech detection.

Haidar et al. [2018] collected a dataset for detecting Arabic Cyberbullying on social media, then trained machine learning classifiers such as Nave Bayes and SVM. They obtained a precision of 90.1 and 93.4% and published another paper using Deep learning where trained simple feed-forward layers architecture, achieving 94.56%.

Albadi et al. [2018] addressed the religious-based hate content problem on social media, in their study they presented a large annotated Arabic dataset, along with collection of lexicon consisting of terms commonly found in religious discussions, for hate speech detection. In their study, the authors trained a Gated Recurrent nets with GRU cells, using pretrained word embeddings to detect religious hate speech with 84% (AUROC). Using the same data, A. G. Chowdhury et al. [2019] proposed ARHNet (Arabic Religious Hate Speech Net) model incorporates both Arabic Word Embeddings and Social Network Graphs for the detection of religious hate speech, their system obtained f1-score of 78%.

Many recent studies have addressed the problem of hate speech in English, Chatzakou et al.

(2019) proposed on a concrete study to understand the characteristics of abusive behavior in Twitter to detect Cyberbullying and cyberaggression in English text. They analyzed 1.2 million users and 2.1 million tweets, comparing users participating in discussions around seemingly normal topics, to those more likely to be hate-related, and also explored specific manifestations of abusive behavior, i.e., cyberbullying and cyber-aggression, in one of the hate-related communities. Using various SOTA machine learning algorithms, they classify these accounts with over 90% accuracy and AUC.

2.5.3 Related works in Arabic Emotion Speech Recognition

The field of Speech Emotion Recognition (SER) has been the subject of numerous studies in various languages. One notable study by Pan utilized MFCC as features and SVM as the classifier, achieving remarkable results with 91.3% accuracy on his Chinese dataset and 95.08% on the EMO-DB dataset (Shen et al. 2011).

A. Khalil et al. 2018 collected an Arabic speech corpus from the TV show "The Opposite Direction" and achieved a 77% classification rate using an SVM classifier. (Al-Faham et al. 2016) also used SVM for their Arabic dataset, resulting in an accuracy of 93.12%.

Harar et al. 2017 utilized a DNN architecture for speech emotion recognition, training the model on the EmoDB dataset (Burkhardt et al. 2005) and achieving an overall test accuracy of 96.97%.

Dealing with Arabic speech published in (Shahin et al. 2019) introduced a hybrid model of Gaussian mode Cheng et al. (2012) and Ayadi et al. (2017) and DNN. This classifier was trained on Arabic United Emirates DB and its performance accuracy was indexed at 83.97%.

J. Kim et al. (2018) proposed a model named *EmNet* that consists of feature extraction, feature normalization, four CNN layers, and two LSTM layers. *EmNet* was evaluated on the EMO-DB dataset and achieved an SER performance of 88.9%.

Y.Li et al. (2019) constructed a CNN-BLSTM architecture (2-layer CNN, 2-layer BLSTM). This model improved classification performance up to 82.8% on IEMOCAP (Busso et al. 2008).

Y.Hifny et al. (2019) implemented two neural architectures to address the problem of emotion recognition from speech. Their first architecture was CNN-LSTM model while the second was

based on the CNN model.

The accuracies obtained on the KSUEmotions dataset (A.Meftah et al. [2014](#)) were 87.2% for the first model and 85% for the second. A recent paper published in C.Li et al. ([2020](#)) used the AMIGOS dataset Correa et al. ([2018](#)) to recognize emotions from natural speech with the LSTM model which achieved an accuracy of 83.3%.

Similarly to our work in Yahia Cherif et al. ([2021](#)), there have been multiple trials for collecting speech corpora for the Algerian dialect, and training multiple machine learning classifiers to propose efficient pipelines for this specific task.

Zantout et al. ([2019](#)) proposed the use of five ensemble classification models including Bagging, Random Forests, and others to enhance the performance of a vocal emotion system. The experiment on the ANAD dataset (Klaylat et al. [2018](#)) resulted in an average prediction accuracy of 95.95%. In a similar study to ours, Dahmani et al. [2019](#) built an automatic speech recognition system for the Algerian dialect using audio signals recorded from the TV show "Red Line." The system was tested using KNN, Adaboost, and Random Forests algorithms, yielding an F1-score of 0.48. Random Forests, KNN, and SVM were also applied to the NATURAL dataset Morrison, R. Wang, et al. ([2007](#)) in a study by Morrison and Silva ([2007](#)), resulting in an accuracy of 76.93% using SVM. The use of ensemble learning methods was explored in another study by Schuller et al. ([2006](#)), using Bagging, Multiboosting, and Adaboosting models yielding accuracy results of 70.7%, 72.5%, and 72.3% respectively on the EMoDB dataset.

Ykhlef et al. ([2019](#)), presented the preliminary results of the building of the first acted Emotional Speech Corpus of Algerian Dialect (*ESCAD*). The corpus consists of recordings of four emotional states (neutral, angry, happy, and disgust) by nonprofessional native speakers. The first version of the corpus includes recordings of angry and neutral states from 53 participants, both male and female. The results were used to identify unskilled speakers, who will not be included in the recording of the remaining emotions. Lately, with the recent breakthrough of the Transformer models, the field of SER has witnessed a new era of effective performance. Multiple studies fine-tuned various BERT models for ESR, like in Lee et al. [2020](#); Lee et al. [2021](#).

Despite the progress witnessed in this area, there is a still much required work to be done to cover more aspects of Arabic SER. Our work in this matter covers multiple points to advance the research field by proposing a new large annotated speech corpora, efficient preprocessing pipelines, and robust classifiers based on a hybrid CNN-LSTM neural networks model.

Table 2.2: SOTA in Arabic Speech Emotion Recognition

| Paper | Model | Dataset | Results |
|------------------------------|---------------------|---------------------|---------|
| C.Li et al. (2020) | LSTM | AMIGOS | 83.30% |
| Y.Li et al. (2019) | CNN-BLSTM | IEMOCAP | 82.80% |
| Y.Hifny et al. (2019) | CNN-LSTM | KSUEmotion | 87.20% |
| Zhao et al. (2019) | CNN-LSTM | EmoDB | 95.73% |
| Zhao et al. (2019) | CNN-LSTM | IEMOCAP | 95.89% |
| Dahmani et al. (2019) | ML-based models | Algerian Dialect DB | 48% |
| Shahin et al. (2019) | NN, Fuzzy Logic | Arabic DB | 94.50% |
| Eljawad et al. (2019) | CNN-LSTM | IEMOCAP | 88.90% |
| Zantout et al. (2019) | SVM, Decision Trees | Arabic DB | 77% |
| J. Kim et al. (2018) | LSTM | IEMOCAP | 64.93% |
| A. Khalil et al. (2018) | DNN | eINTERFACE'05 | 84% |
| Latif et al. (2017) | DNN | EmoDB | 96.97% |
| Hu et al. (2007) | SVM | AES database | 93.12% |
| C. W. Huang et al. (2017) | SVM | EmoDB | 95.30% |
| Harar et al. (2017) | SVM | - | 82.50% |
| Al-Faham et al. (2016) | SVM | NATURAL | 76.93% |
| Shen et al. (2011) | Adaboosting | EmoDB | 72.3% |
| | Bagging | | 72.5% |
| Morrison and De Silva (2007) | ANN | - | 50% |
| Schuller et al. (2006) | ANN | - | 65% |

2.5.4 Related Works in Arabic Dialect Identification

recently more research focused on the identification of different Arabic dialects on social media, as well as the collection of data from various regions in the Arab world conveyed in both Arabizi and Arabic script. (Sayadi et al. 2016) provided a manually annotated dataset with almost 50,000 tweets from 8293 users, then studied sentiment analysis on Tunisian dialect and Modern Standard Arabic.

Tobaili 2016 annotated a corpus of the splitTwitter data stream coming from within Lebanon and Egypt, where users speak Araby-Englizi, then trained a classifier and achieved an average classification accuracy of 93% and 96% for Lebanon and Egypt datasets respectively. I. Guellil et al. 2017 proposed an approach for Arabic dialect identification in social media, specifically the Algeria dialect. The authors applied their approach to 100 messages manually annotated, and they achieved an accuracy of more than 60%.

D. Seddah et al. 2020 introduced the first treebank for a romanized user-generated content variety of the Algerian dialect, as mentioned in their paper. The content written in the Arabic language on the Internet is characterized by a high degree of linguistic diversity due to the

use of colloquial dialects and writing in Roman characters, in addition to the phenomenon of code-switching. In addition to the annotated data, the authors provide around 1 million tokens (over 46k sentences) of unlabeled Arabizi content. Khaled Darwish [2013] addressed the problem of identifying Arabizi (Arabic text written with Latin characters) using word and sequence-level features achieving 98.5%, then converting it into Arabic characters using transliteration mining with language modeling achieving 88.7%

Many studies also focused on the collection of Arabizi and different Arabic dialects corpora from social media, Zaidan et al. (Zaidan et al. [2014]) collected a corpus, from three Arabic newspapers of Levantine, Gulf, and Egyptian dialects.

Cotterell et al. [2014] also presented extensive dialectal data from online resources for Algerian, Egyptian, Iraqi, and Gulf.

With the arrival of recent pretrained large language models (LLMs) supporting the Arabic language, multiple trials have been put through to test the efficiency of these models for the differentiation of multiple Arabic dialects. As an example, multiple shared-tasks and competitions were organized to evaluate these LLMs. The Nuanced Arabic Dialect Identification (NADI) shared tasks (Abdul-Mageed, C. Zhang, AbdelRahim Elmadany, et al. [2022])³⁴ aims to identify dialects at the provincial level and represents the inaugural effort to focus on sub-country level dialects in a naturalistic setting. The corpus encompasses data collected from 100 provinces across all 21 Arab nations and originates from the Twitter platform.

The MADAR Shared Task (Bouamor, Hassan, et al. [2019]) is a challenge for Fine-Grained Dialect Identification in the Arabic language. The task data was generated as part of the Multi-Arabic Dialect Applications and Resources (MADAR) project. A total of 21 teams from 15 nations participated in the shared task.

The Talafha et al. [2020] team ranked first in the NADI shared task competition (Abdul-Mageed, C. Zhang, AbdelRahim Elmadany, et al. [2022]). Their system is composed of three main stages: Initially, the team pre-trained an Arabic-BERT model using the 10 million tweets provided by the NADI competition organizers. Then, they trained the pre-trained model on the NADI labeled data for Task 1 multiple times using different combinations of maximum sentence length and learning rate. To determine the final prediction for a tweet, they selected the 4 best-

³<https://nadi.dlnlp.ai/>

⁴<https://sites.google.com/view/nadi-shared-task>

performing iterations from these trials, based on their performance on the development dataset, and combined their softmax predictions through an element-wise averaging function.

Table 2.3 summarizes some of the datasets and shared-tasks for Arabic dialect identification.

Table 2.3: Summary of some available Arabic dialects datasets

| Corpus Name | Reference |
|-------------|--|
| ADI | Bouamor, N. Habash, et al. (2018) |
| MPCA | Malmasi et al. (2016) |
| AlgDI | Imène Guellil et al. (2016) |
| MADAR | Bouamor, N. Habash, et al. (2018) |
| NADI2020 | Abdul-Mageed, C. Zhang, Bouamor, et al. (2020) |
| NADI2021 | Abdul-Mageed, C. Zhang, AbdelRahim Elmadany, et al. (2021) |
| ADI17 | Shon et al. (2020) |
| QADI | Abdelali, Hamdy Mubarak, et al. (2021) |
| NADI2022 | Abdul-Mageed, C. Zhang, AbdelRahim Elmadany, et al. (2022) |

2.5.5 Related works in Arabic Language Models

Recently, there has been a surge in the development of SOTA Pretrained Language Models (PLMs) which have revolutionized the NLP field. These models are trained on a large corpus of text data and can be fine-tuned on various NLP tasks like sentiment analysis, named entity recognition, and machine translation, among others.

In this section, we present some of the recent SOTA PLMs and also discuss some models that have been developed specifically for the Arabic language.

For the past decade, word embeddings were the standard method for representing text sequences (Tomas Mikolov, Sutskever, et al. 2013; Pennington et al. 2014). However, they have a major constraint since each word can only have a single vector representation. Nevertheless, words can have multiple meanings depending on their context, especially in rich morphology languages like Arabic. Therefore, recent work focused on context-dependent representations (Peters et al. 2018; Devlin et al. 2019; Zhilin Yang et al. 2019; Brown et al. 2020; Lan et al. 2019) pushing the sense of transfer learning in NLP even further (Ruder et al. 2019).

Although their remarkable results on downstream tasks are well-acknowledged, these models require large amounts of raw text and a lot of computational resources to be pre-trained, which limits their availability to high-resource languages. Even multilingual models (Devlin et al. 2019; Conneau et al. 2020), which are usually trained for several hundreds of languages, often

standard Arabic was considering, ignoring the variety of Arabic dialects.

Distributed Text representation is a form of representing text sequences as vectors before passing them into neural network model for training. For the past two decades, several word representation have been proposed; word embeddings are breakthrough models in the field of NLP where they represent words by fixed vectors, followed by Word2vec (Tomas Mikolov, Sutskever, et al. 2013) and Glove (Pennington et al. 2014) models that encodes the word in sort of meaningful vectors (similar words have similar vector representations), these models have brought significant improvement in many NLP tasks, yet they come with major constraints is that each word in the vocabulary can only have a single vector representation, this affects the case where words could have multiple meanings depending on the context it is being used, especially in rich morphology language like Arabic.

Therefore, recent works focused on context-dependent embeddings, meaning a word could have flexible word representation according to the context in it is employed. ULMfiT (Howard et al. 2018) and Embeddings from Language Models (ELMO) (Peters et al. 2018) were the first models that brought the sense of transfer learning into NLP (Ruder et al. 2019), followed by the arrival of OPENAI GPT (Brown et al. 2020), BERT (Devlin et al. 2019), XL-NET (Zhilin Yang et al. 2019), T5 (Zeng et al. 2018) and so many other BERT-optimized versions language models such Xlm-RoBERTa (Y. Liu et al. 2019) and ALBERT (Lan et al. 2019).

These models are first pretrained on a large corpus of text data and then fine-tuned on downstream tasks like named entity recognition and sentiment analysis which proven to add substantial improvements on the performance compared to classical task-specific approaches.

Although their remarkable results on downstream tasks, these language models require large amounts of raw text and a lot of computational resources needed for the pretraining (Zhilin Yang et al. 2019), which limited their availability only for rich resource languages such as English. To overcome this, several multi-lingual models mBERT (that is pretrained on a corpus of 104 languages), and XLM-RoBERTa (pretrained on 100 languages) were pretrained on hundreds of languages, yet these models were not “*enough*” for many other languages like Arabic (Antoun et al. 2020).

Besides, most of the proposed models have been trained on standard Arabic. The first released transformer-based model for the Arabic language was AraBERT (Antoun et al. 2020) which has been pre-trained on 23 GB of MSA extracted from Wikipedia, news articles, etc. It was followed by ARBERT (Abdul-Mageed et al. 2021) which was pre-trained on a larger MSA dataset (61 GB from Wikipedia, news articles, books, etc.). The same authors also trained MARBERT on 128 GB of various Arabic dialects (1 billion tweets). In another work, Abdelali, Hassan, et al. 2021 pre-trained QARiB on a collection of 420 Million tweets in dialectal Arabic and 180 Million sentences in standard Arabic. Finally, Inoue et al. 2021 released CamelBERT which was pre-trained on four types of Arabic datasets: Dialectal Arabic (54 GB), MSA (107 GB), Classical Arabic (6 GB) and a mixture of the last three datasets (167 GB).

However, as was the case of inadequacy for the multilingual mBERT model on Arabic, these new Arabic models are not also enough for several Arabic dialects such as the Algerian dialect, due to the difference between the morphological syntax and the complexity of the Algerian Dialect.

The multilingual models mBERT and XLM-R, along with the Arabic language models were trained on Arabic text corpora targeting mostly MSA, and due to high differences between MSA and Dialectal Arabic, the need for a monolingual dialect-level language model is needed.

Table 2.4: Recent pretrained language models for the Arabic language (AbdelRahim Elmadany, Nagoudi, et al. 2022)

| Model | Text type | Corpus size | # tokens | Tok. Size | #Params |
|---------------|-------------|-------------|----------|-----------|---------|
| ARBERT | MSA | 61GB | 6.2B | 100K | 163M |
| ARBERTv2 | MSA, DA | 243GB | 27.8B | 100K | 163M |
| MARBERT | MSA, DA | 128GB | 15.6B | 100K | 163M |
| MARBERTv2 | MSA | 198GB | 21.4B | 100K | 163M |
| AraBERT | MSA | 27GB | 2.5B | 64K | 135M |
| AraELECTRA | MSA | 77GB | 8.8B | 64K | 135M |
| ArabicBERT | MSA | 95GB | 8.2B | 64K | 135M |
| Arabic-ALBERT | MSA | 33GB | 4.4B | 32K | 110M |
| QARiB | MSA, DA | 97GB | 14B | 64K | 135M |
| CAMeLBERT | MSA, DA, CA | 167GB | 8.8B | 30K | 108M |

2.5.6 Related works in Text Classification

Working with Arabic text is more challenging than working with English text because of inherent language characteristics and lack of efficient resources. For instance, there is a significant discrepancy in vocabulary size (around 12.3 million words compared to 600,000 words in English) and in use of diphthongs and long vowels as in-fixes. Besides, from sentence structure perspective, Arabic has verbal and nominal sentences where the latter do not require a verb, which adds extra difficulty to parsing disambiguation in text classification tasks. Besides, the nature of classification, which may involve short versus long documents, multiclass versus binary classification, or use of multi-label classification plays also a key role in enforcing a given methodological framework. This explains why Arabic text classification research has been conducted closely with the availability of benchmarks and corpora data as highlighted in survey papers (Sayed et al. [2019](#); Al Sbou [2018](#); Alammery [2022](#)).

Indeed, El Rifai et al. [2021](#) collected an Arabic news dataset that contains 90k single-labeled articles and a 290k multi-labeled news dataset. The authors trained multiple classical ML classifiers using TF-IDF features and 10 (ten) deep learning models. Khoja et al. [2017](#) presented a dataset that consisted of 30K pairs of news headline documents trained using a Seq2seq and attention models for text generation purposes. Similarly, Alotaiby [2011](#) trained a Character Cross-Correlation model for Arabic news generation on 260 Arabic news documents.

Alzawaydeh et al. [2018](#) presented a study on metaphor detection on Arabic and English sports news headlines. Their study identified commonalities between Arabic and English classification schemes based on metaphorical concepts. For this purpose, the authors collected a corpus of 400 Football news headlines (200 for English and 200 for Arabic).

Haider et al. [2019](#) analyzed a large Arabic and English news corpora mentioning the Lybian regime system in the period between 2011 and 2012 to see if the analysis of news headlines could yield similar insights to full news documents. Their research outlined that news headline analysis corresponds to a good down-sampling analysis and can help to reduce the complexity of tackling large news documents.

Using Khalij subset from SANAD dataset (Muaad et al. [2022](#)), trained multiple classifiers such as LinearSVC, and logistic regression using BoW and TF-IDF features and compared their

performance using a CNN model. The CNN model outperforms classical ML models in this subset data. Elsayed et al. [2020] presented Arabic news headlines labeled for emotion recognition. The dataset collects 1698 documents labeled for seven emotional categories. The authors then trained a two-level CNN model using N-grams features to achieve a result of 90% in classification accuracy level.

Mohammed et al. [2019] collected an Arabic news headlines dataset that includes 2999 headlines labeled as sarcastic and another 2999 labeled as normal from two Arabic news websites. The authors afterward conducted several experiments using deep learning models to classify the headlines as either sarcastic or normal. Their CNN model reported the best classification accuracy up to 92%.

Table 2.5: Some recent works on Arabic document classification

| Paper | Dataset | Method |
|-------------------------------|--|------------------------------|
| El Rifai et al. [2021] | 90k single-labeled articles | RNN |
| El Rifai et al. [2021] | 290k multi-labeled articles | RNN |
| Khoja et al. [2017] | 30K pairs of news headline | Seq2seq and attention models |
| Alotaiby [2011] | 260 Arabic news documents | Character Cross-Correlation |
| Alzawaydeh et al. [2018] | 400 Football news headlines multiclass | RNN based models |
| Haider et al. [2019] | headline multiclass | EDA |
| Elsayed et al. [2020] | 1698 Arabic headlines multiclass | N-grams |
| Mohammed et al. [2019] | 2999 headlines multicalss | CNN |
| Einea et al. [2019] | 181,152 multiclass | CNN and RNN |
| Einea et al. [2019] | 35,416 multilabel | CNN and RNN |
| S. A. Chowdhury et al. [2020] | 10k multiclass | Transformers |
| Muaad et al. [2022] | Khalij from SANAD | CNN |
| Boukil et al. [2018] | 111k news documents | CNN |

In general, due to Arabic language rich morphology (N. Habash [2007]) and level of complexity (N. Y. Habash [2010]; S. Khalifa et al. [2020]), the research community leverage deep learning models such as CNN (Alshaalan et al. [2020]), Transformers (S. A. Chowdhury et al. [2020]), RAE (Al Sallab et al. [2015]), BERT (Antoun et al. [2020]) as they enable language processing efficiently. Although, in short text-classification task, (Zhan et al. [2017]) reported that many deep learning models yielded inappropriate results, likely due to data sparsity. Recently, multiple language models have been proposed such as AraBERT (Antoun et al. [2020]), DziribERT (Antoun et al. [2020]) and BAERT (Younes et al. [2020]). Fine-tuning these language models has reported remarkable performance on multiple downstream tasks such as sentiment analysis (Farha et al. [2021b]; Xu et al. [2019]; Chouikhi et al. [2021]; Antoun et al. [2020]), text classification (El-Alami

et al. [2021]; Hammoud et al. [2021]; ElJundi et al. [2019] and question answering (Mozannar et al. [2019]; Alwaneen et al. [2022]; Ngai et al. [2021]). Finally, In the context of news and documents classification and categorization, several SOTA datasets are worth noticing as they significantly contributed to short-document and multi-label classification approaches:

- Arabic Social Media News Dataset (ASND) (S. A. Chowdhury et al. [2020]): the dataset was collected using Aljazeera (popular Arabic news media channel) account on social media and Youtube, where a total of 10k posts were labeled into 12 categories.
- Large Single Labeled Arabic News Dataset (SANAD) (Einea et al. [2019]): this dataset groups 195k news splitted into 6/7 classes scraped from various Arabic online media.
- Multi-label Arabic News Dataset (NADIA) (Einea et al. [2019]): The Nadia dataset was collected from two Arabic news websites, Skynews and Masrawi, and consists of two collections of news articles. The collection from Skynews consists of 35,416 news articles grouped into 24 categories, and the collection from Masrawi consists of 451,230 news articles belonging to 28 classes. In our experiments, we use only the Skynews collection, where each news article can have up to 8 labels, as illustrated in Figure 2.4. The average length of the news articles in the Skynews dataset is 260 tokens.
- Other dataset (Boukil et al. [2018]): is a collection of 111,728 Arabic news documents, grouped into five unbalanced classes (sport, politics, culture, economy, and diverse news) the dataset was collected from three online news media and was used in (Madhfar et al. [2019]) using several machine learning models using Bag of words features.

In SANAD and NADIA datasets, the number of words per document (news length) is ≈ 370 words per document, thereby, the need for a large balanced Arabic *short text* dataset is still missing. Likewise, News headline categorization, which is linked to short text classification tasks (Mohammed et al. [2019]) remains challenging for many languages (M. Chen et al. [2011]), including the Arabic language (Khoja et al. [2017]), which calls for further research in this issue.

As a contribution to proposing a new data source for the NLP community, We introduce a new large Arabic headlines dataset that is made publicly available to the research community used to validate our developed model. At the same time, for comparison purposes, we evaluated multiple deep learning models on large Arabic text classification benchmarks using different word embeddings and pretrained language models.

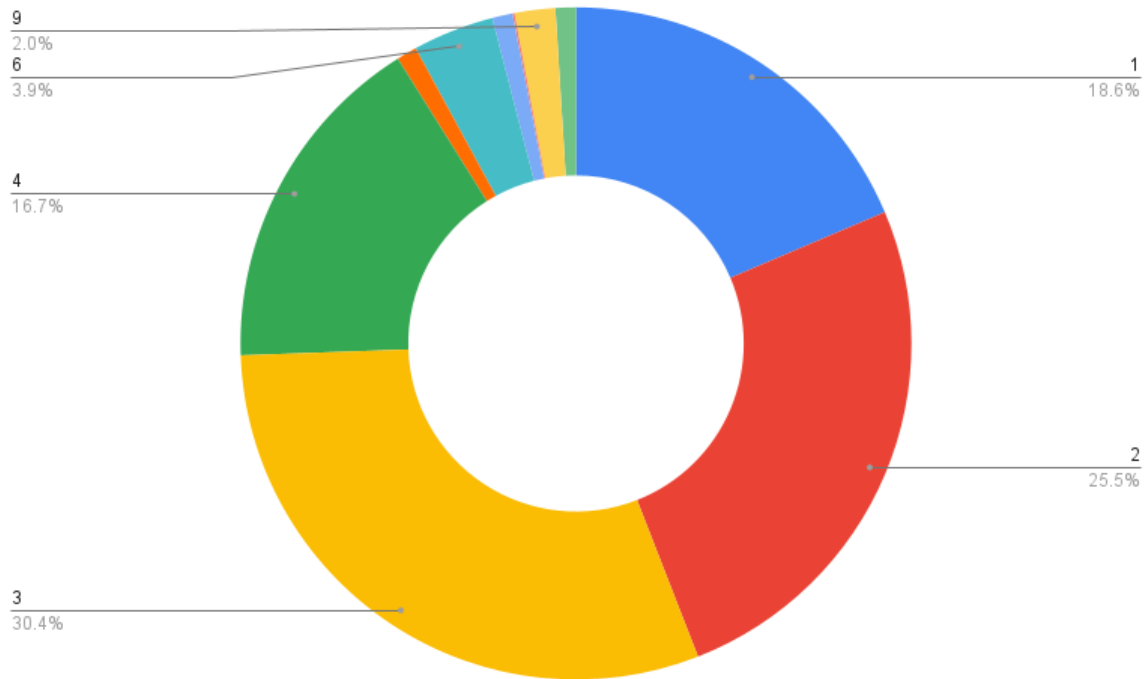


Figure 2.4: Class distribution of the multi-class NADiA dataset.

2.6 Conclusion

In this informative chapter, we delved into the complexities of the Arabic language and the linguistic challenges it poses. We explored how recent advancements in deep learning models have helped to overcome some of these challenges, leading to exciting progress in the field.

Furthermore, we discussed related works that have been divided into various sections based on their respective domains. By presenting this comprehensive overview, readers can gain a deeper understanding of the unique characteristics of the Arabic language and appreciate the innovative approaches that have been developed to tackle its challenges.

Chapter 3

Arabic text and speech analysis

In this chapter we tackle three domain areas of analyzing Arabic text and speech using various deep learning techniques. We present our contributions to diverse areas within the scope of text and speech analysis. In the first section, we present our research efforts focusing on Arabic sentiment analysis (ASA), for which we have carried out a comprehensive literature review and developed new deep learning models. Additionally, we have analyzed various word embedding techniques and evaluated their effectiveness. Secondly, we delve into the area of inappropriate speech detection in Arabic text, utilizing deep learning techniques to train and compare a range of models. Then introduce a novel offensive speech dataset for the Algerian dialect and analyze it, incorporating new advancements in the field of NLP. In the last section, we present our work on the analysis of emotions conveyed in speech data in the Algerian dialect. To this end, we collect and annotated a new audio dataset, and trained and evaluated multiple deep-learning models.

3.1 Arabic Sentiment Analysis

3.1.1 Introduction

In the era of social media, sentiment analysis emerged as an active research area in natural language processing. SA aims to identify and monitor sentiment polarity/strength of user-generated text that encapsulates his/her opinion, emotion, and/or attitude towards entities such as services, organizations, products and events, among others. The interest in SA grew with the exponential increase in the recorded number of opinionated data in blogs and news.

This renders SA analytics central in many disciplines and organizations.

For instance, nowadays, many customers consult users' comments and sentiment polarity before making a purchasing decision on a newly introduced product or service. Organizations use SA either as a substitute or a supplement to standard surveys and opinion pools performed using questionnaire and field observation methods. Consequently, SA provides organizations and policy-makers with valuable insights to reshape their policy, business, marketing, and/or communication strategies in a way to accommodate citizens' concerns, increase their organizational efficiency and create a societal impact. This makes SA almost a necessity in various disciplines where capturing citizen's views is deemed important (B. Liu 2016). This ranges from traditional computer science fields to humanities and medical fields. This is because the study of opinion trends becomes central to almost all human activities whenever there is a need to make a decision through either a semi-automated or a fully manual process. The variety of electronic documents used as input for SA, in terms of their structures, access policy, semantics and content structure, among others, creates continuous challenges to automatic sentiment analysis tasks.

Among these challenges, one distinguishes language-related patterns. Indeed, the efficiency of the existing NLP parsers differs from one language to another. This is due to the quality of the training dataset and methodological approaches employed in the development of the underlined parser as well as the complexity of the social norms and stylistic cues embedded in the language. On the other hand, a simple statistical count of publication numbers per language indicates clearly that the Arabic language is underrepresented, see Fig. 4.1. This also holds when comparing the maturity of the technology and the level of performance obtained. For instance, the best state-of-the-art sentiment polarity accuracy achieved in competition (Rosenthal et al. 2017) associated with Arabic SA is only 58.1% as compared to more than 96% in (Al-Dabet and Tedmori 2019)!

This motivates the current research which aims to investigate a new deep learning approach for Arabic sentiment analysis tasks. Loosely speaking, Arabic language groups cover nearly 500 million speakers worldwide (Boudad et al. 2018), making it the fourth common spoken language (Imane Guellil, Saadane, et al. 2021) and the largest member of the Semitic Language Family. Besides, the Arab world has recently witnessed a series of events (e.g., Arab Spring) and fast-growing e-commerce trading activities in the area as well as the proliferation of social

media users, which raise the prospects and the importance of Arabic sentiment analysis.

Technology for unfolding Arabic sentiment analysis is ultimately linked with that of sentiment analysis and natural language processing at wide regardless of the language context. In this context, one distinguishes at least three streams of approaches: machine learning (both supervised and unsupervised), lexicon-based and hybrid methods (L. Zhang et al. 2011). Supervised machine learning approaches attempt to use appropriate feature engineering methods (e.g., tf-idf features, N-grams, selected adjective/adverbs, specialized dictionaries, . . .), a given classifier (e.g., Naives Bayes' (NB), Support Vector Machine (SVM), Linear Regression (LR), . . .) and appropriate training/testing dataset. Lexicon-based methods use a collection of sentiment terms that are precompiled into a sentiment lexicon. This is further divided into the dictionary- and corpus-based approaches that use either semantic or statistical methods to gauge the extent of a given sentiment polarity by accounting for various grammatical constructs and syntactic patterns. While hybrid approaches involve combining machine learning and lexicon-based approaches.

A special case of machine learning-based methods, which has shown state-of-the-art results in many SA-related competitions is the deep learning methods. For instance, all the top winners in recent SemEval Sentiment Analysis competitions used deep learning models (e.g., CNN, RNN, LSTM) (Rebiai et al. 2019).

Recent studies on deep-learning Arabic sentiment analysis have focused on using RNN (Heikal Maha 2018; Al-Azani et al. 2018) that are specialized in processing sequential data. Nevertheless, a major limitation of RNNs is their incapacity to memorize longer sequences and the vanishing and/or exploding gradient estimate (Bengio et al. 1994), which restrict their capability to account for the discourse effect. RNN with memory gates such as LSTM (Hochreiter et al. 1996), GRU (Chung et al. 2014), and Bidirectional RNN (Schuster et al. 1997) are often put forward as alternative strategies to tackle this issue. Still, their capability of memorizing long sequences is also questioned because the gated networks' sentence representation depends only on past and current data states, which is not the case for most sentiment analysis problems, especially in the Arabic SA context. While BiRNN is found to suffer from sequential bias as well as a lack of interpretability, Attention Mechanism (Bahdanau et al. 2014a) was introduced to allow RNNs to focus on the input sequence's relevant segments. Its performance is found to surpass that of the recurrent network model in memorizing longer sequences. Nowadays attention

mechanism plays a dominant role in most NLP tasks, especially in the many state-of-the-art models like the Transformer (Vaswani et al. 2017b), BERT (Devlin et al. 2018), GPT (Radford, J. Wu, et al. 2019), and XL-NET (Zhilin Yang et al. 2019). Beyond improving the neural nets' performance, the attention mechanism brings an interpretable aspect to neural network-based models.

We, therefore, hypothesize that the contribution of the attention mechanism to Arabic SA tasks can be positive as well. In this chapter, we thereby advocate a new deep learning model that uses the attention mechanism as an additive layer to the BiGRU model. The proposed model is next tested using three Arabic labeled dataset: LABR (Large-scale Arabic book review) (Aly et al. 2013), HARD (hotel Arabic review dataset) (Elnagar, Y. Khalifa, et al. 2018), BRAD (Book review in Arabic dataset) (Elnagar and Einea 2016). Furthermore, the generalization of the model beyond the SA tasks is investigated by testing the model on other NLP tasks; mainly, a hate speech dataset and a Russian SA dataset.

Research Objectives Due to the relatively low accuracy and the inherent limitation of tools employed in Arabic language analysis such as morphological analyzers, PoS taggers and Stemmers, there is a potential for systems that can automatically extract and classify opinions present in user-generated documents. This chapter aims to contribute to this overall goal. More specifically, the proposed research objectives are to:

- Comprehensively review the challenges associated with Arabic SA in both MSA and DA,
- Review deep learning approaches for Arabic SA.
- Propose and implement a new additive-sequence level model for Dialect Arabic sentiment polarity detection,
- Propose an approach for tuning the parameters of the developed model.
- Demonstrate the feasibility of the proposal through comparison with other state-of-the-art models.
- Demonstrate the extension of the models to other applications.

Contributions The main contributions of this section are summarized below:

- We trained both BiLSTM and BiGRU models on four datasets, by performing a grid search to select the best hyperparameter set and select the winning architecture to train the proposed model, using two sets of embeddings: Learnable embeddings and FastText embeddings.
- We proposed a BiGRU additive-attention sequence-level model to detect and analyze sentiments from Arabic reviews. Then we tested its performance on three supervised Arabic datasets labeled as Positive and Negative.
- We experimentally verified that our proposed model outperformed the baseline and some state-of-the-art models, and demonstrated that our model is language-independent by testing it on English and Russian datasets.

The chapter is organized as follows. Section 2 outlines the challenges associated to natural language processing of Arabic language. In Section 3, we introduce the most recent related works targeting the Arabic sentiment analysis problem. Section 4 highlights the general methodology, including the motivation grounds, the overall architecture, and its different components. Section 5 emphasizes the experimental setting and the associated results, highlighting the dataset employed, the tuning of hyper-parameters of the model, the baseline models, the performance metrics, and the obtained results. Finally, Section 6 discusses the implications of the results, and inherent limitations. Section 7 summarizes the major findings and lists the perspective works.

3.1.2 Proposed system

Motivation

The proposed system builds on the merits of deep learning methods as the currently acknowledged state-of-the-art approach for sentiment analysis as pointed out in the previous section. More specifically, our approach advocates the use of the attention mechanism, which becomes one of the core technologies in deep learning after its huge success in neural machine translation (Zichao Yang et al. [2016](#)). This follows the intuition of human visual attention where the user can focus on a certain region of an image with 'high resolution' while perceiving the surrounding image in 'low resolution', and then adjusting the focal point over time. The main motivations for the choice of attention mechanism in our proposal are fourfold. First, the inherent property of the attention mechanism to focus on most of the salient parts of the input-space, instead of

encoding the full sentence length as in typical RNN architectures, can substantially enhance the optimization performance of the deep learning model.

Second, given the sparse structure and the complexity of the Arabic language morphology and its semantics, as pointed out in the introduction section of this chapter, any incremental gain in optimization will be very welcomed and crucial for the overall system performance.

Third, the attention weights can be used as a tool to interpret the behavior of the associated neural network architecture, which is notoriously difficult to comprehend, and, thereby, add an interpretability dimension to the underlined neural architecture. Fourth, the success of attention-mechanism architectures in sentiment analysis in various languages (Zichao Yang et al. 2016), including Arabic language (Al-Dabet and Tedmori 2019) provides a good indicator of the potential merits and promises of such research direction.

Reviewing the existing architectures applied to Arabic sentiment analysis using the attention mechanism-based approach, one distinguishes the deep attention-based review level sentiment analysis put forward in (Almani et al. 2020). Their model uses a multi-layer architecture in the following way. First, the embedding layer passes the distributed word representation of the input textual review to the GRU-based layer to produce a hidden review representation. Second, a soft attention layer is embedded on the top of the GRU layer to perform the sum of GRU hidden representations according to the generated weights of each word and output a distributed vector representation of the input review according to the identified salient words. Third, the review vector representation outputted by the attention layer is fed to a fully connected sigmoid logistic regression layer to generate the final polarity classification. In parallel, (Al-Dabet and Tedmori 2019) used a three-layer attention mechanism architecture where the first layer corresponds to the word-embedding created using a Wikipedia dataset whose vector representation is fed to a simple recurrent model, which utilizes LSTM and Gated Recurrent Units (GRU). The output of the recurrent model is then fed to the attention layer, which produces, for each input sentence, a weighted sum of the attention weights and the words' hidden vectors, which are then passed to a sigmoid layer that performs the binary classification (positive versus negative polarity) task.

In comparison to the above architectures, our model is rather close to (Zichao Yang et al.

[2016]) where a Bidirectional GRU reads the individual sentence input from left to right and vice versa to capture the contextual information, and whose results are then concatenated. Besides, for learning long and short dependencies, we used two sets of recurrent gated networks; namely, LSTM and GRU, where an empirical approach was employed to choose between the LSTM or GRU layer according to their training performance on some experimental data. The motivation for using a such training approach is rooted back in the well-known vanishing and exploding gradient phenomenon when using backpropagation with RNN structures. At the input embedding layer, we used two types of embeddings. The first one consists of the FastText embedding, which enables us to overcome the difficulty of out-of-vocabulary observed with commonly employed word distributional representation (Tomas Mikolov, K. Chen, et al. [2013]; Tomas Mikolov et al. [2013]).

The second one uses a learnable embedding approach that is inferred from the training samples and varies at each time increment. The motivation for doing so is to capture the increasing variations of Arabic language structures when dealing with sentiment as well as the potential limitations of the pre-trained models employed in generating FastText. Similarly to (Al-Dabet and Tedmori [2019]), we also employed a sigmoid layer to perform the binary classification (positive/negative polarity). The gradients of our model are backpropagated through both the Bidirectional recurrent net and the attention blocks, making the different parameters updated at each iteration. Furthermore, the model takes special care of the preprocessing stage where noisy terms can play a central role in guiding the sentiment polarity. For instance, removal of negation characters can turn a negative (resp. positive) statement into a positive (resp. negative) statement. The next subsection details the different components of the architecture of our model.

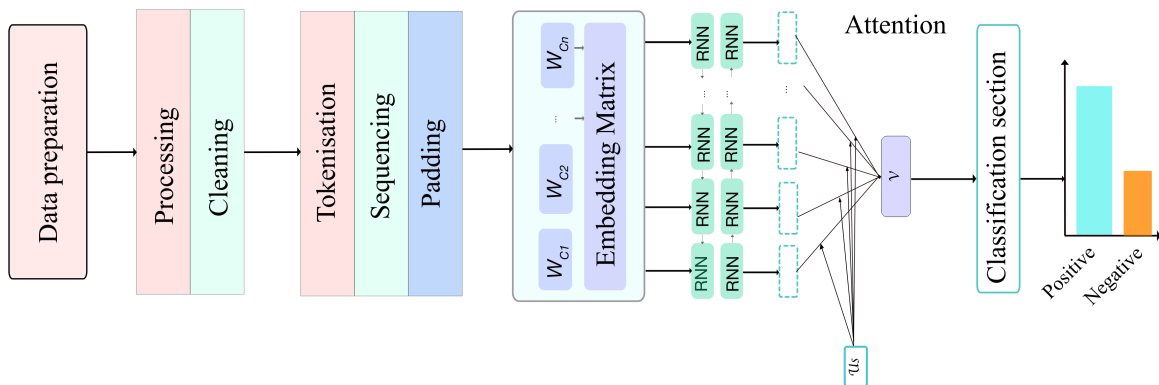


Figure 3.1: Generic flow graph of the proposed system.

Overall architecture

Fig. 3.1 summarizes the main components of our architecture. After a preprocessing stage and appropriate data representation, the architecture includes an encoding section, which encapsulates the various embedding whose outcome is fed to a bidirectional RNN (BiLSTM and BiGRU) layer. The latter is then fed to the attention layer, which, in turn, is fed to the classification module that assigns contextual representation of the input sequences to their corresponding classes using a sigmoid activation function.

More formally, the input sequences are tokenized, padded according to the value of MaxLen (maximum sentence length) to ensure coherence, then passed to the embedding layer. This generates embeddings for each received token, which are then inputted to the Bidirectional RNN with a forward and backward pass to effectively capture and represent the information context. The encoded vectors are then passed to the attention layer. The latter assigns a weight to each token according to its contribution to a given sentiment polarity class. The attention layer's output is then passed to the feedforward layer, which, in turn, passes it to the sigmoid layer that assigns each sequence to its corresponding class.

Detailed architecture

Preprocessing and normalization Data collected from social media and online blogs are often accompanied by unwanted characters due to the presence of miss-spelling, URLs and hash-tags, among others, which can negatively impact the output of the natural language processing modules. This holds mainly because the used dataset is issued from both MSA and Arabic dialects, which contain many unstructured and noisy constructs. Therefore, the pre-processing stage contributes to reducing the impact of such implicit noisy representation. After conducting several trials, we found that the following pre-processing pipeline helped us to enhance both the quality of the training of the employed deep learning models and the accuracy result, while it reduced the vocabulary size:

- Remove unstructured and unmatched Arabic diacritics terms,
- Remove URLs, hashtags, and special characters.
- Remove both Indian and Arabic digits,
- Remove (after two successive occurrences) characters that have more than two successive

repetitions within the same sequence as in جمبييل becomes جميل,

- Use classical MSA structure to normalize unstructured Arabic letters often employed in DA (e.g., normalize اأا by replacing it by the unique form (Alif) ا).

Encoder module As a first step, we obtain the embeddings of the different tokens in each corpus. In our study, we used two sets of Embeddings. The first one referred to *learnable embeddings* consists of word embeddings learned from the training data, while the second one consists of *FastText* pretrained embeddings (Tomas Mikolov, Grave, et al. 2018) trained using CBOV model on a large Arabic corpus from Wikipedia where the surrounding contexts are used to predict the target word.

More specifically, for the learnable embeddings, we used Keras Embedding layer [1](#) where the embedding layer is initialized with random weights, which are then updated with the learning process and the gradient descent for all words in the training dataset. Our implementation assumes a maximum of 150 tokens per post. Padding strategy was then been employed to accommodate the discrepancy of the size of individual posts. We also set the size of the generated embedding vector to 150 per individual token.

From an implementation perspective, in the case of FastText embeddings, we created the vocabulary index of each dataset (detailed in the next section), then each token index X_i is mapped to the associated FastText embedding vector. At the same time, we prevent the model to update the embeddings matrix while training.

The embeddings (both FastText embeddings and trainable embeddings) are therefore passed to the Bidirectional recurrent network block to obtain the annotation of the embedded tokens.

In our study, we used two sets of recurrent gated networks: Bidirectional Long-short Term Memory (BiLSTM) and Bidirectional Gated Recurrent Units (BiGRU) to learn long and short dependencies. Specifically, a simple heuristic voting strategy was employed between BiGRU and BiLSTM for each input sequence. This voting scheme relies on the training performance and the performance obtained by the BiLTM and BiGRU alone, which are also employed as baseline classifiers, for the same input sequence. A such voting scheme, although simple, can be rooted back to the theory of dynamic selection of classifiers, see (Alceu S. Britto 2014) for

¹<https://keras.io/layers/embeddings/>

an overview.

Loosely speaking, this follows the spirit of the majority voting classification scheme of multi-classifiers (Kuncheva [2004](#)) where BiLSTM and BiGRU act as individual classifiers and the information about accuracy and epoch training time serve as a tool to determine the weight of the classifier. Therefore, if the training accuracy of BiLSTM (resp. BiGRU) is more significant than that of BiGRU (resp. BiLSTM), then the weight of BiLSTM is deemed more important. Otherwise, if the training accuracy of the two classifiers are close to each other, then the classifier who has the smallest training time is deemed favorable.

Modelling More formally, at a given time stamp t , given an input text sequence $X^t = (X_1^t, X_2^t, \dots, X_n^t)$, the Bidirectional recurrent network block uses two parallel RNN layers that process the output of the embeddings (context vector) from left to right $\vec{H}^t = (\vec{H}_1, \vec{H}_2, \vec{H}_3, \dots, \vec{H}_n)$ and from right to left $\overleftarrow{H}^t = (\overleftarrow{H}_1, \overleftarrow{H}_2, \overleftarrow{H}_3, \dots, \overleftarrow{H}_n)$, making the model able to use context from previous and later timesteps such as $H^t = (H_1, H_2, \dots, H_n)$ where $H_i = [\vec{H}_i, \overleftarrow{H}_i]$.

In short, H^t summarizes the neighbor posts (sentences) around the post encapsulated by the input sequence X^t . Linking the current state to the previous state, the BiLSTM processes the encoded word vector as follows, assuming an input sequence of fixed size n):

$$\vec{H}_{i,LSTM} = \overrightarrow{LSTM}(X_i^t, \overrightarrow{H}_{i-1}), i \in [1, n] \quad (3.1)$$

$$\overleftarrow{H}_{i,LSTM} = \overleftarrow{LSTM}(X_i^t, \overleftarrow{H}_{i-1}), i \in [1, n] \quad (3.2)$$

The same reasoning applies to the BiGRU layer, when BiGRU encoding was employed instead of BiLSTM:

$$\vec{H}_{i,GRU} = GRU(X_i^t, \overrightarrow{H}_{i-1}), i \in [1, n] \quad (3.3)$$

$$\overleftarrow{H}_{i,GRU} = GRU(X_i^t, \overleftarrow{H}_{i-1}), i \in [n, 1] \quad (3.4)$$

The output is therefore constituted of the concatenation of the two layers (forward and backward

layers).

$$H_{LSTM} = [\vec{H}_{LSTM}, \overleftarrow{H}_{LSTM}] \quad (3.5)$$

$$H_{GRU} = [\vec{H}_{GRU}, \overleftarrow{H}_{GRU}] \quad (3.6)$$

Attention block The attention mechanism used in our study is mainly inspired by the hierarchical attention architecture proposed in (Zichao Yang et al. 2016) that has a two-level hierarchical structure: word-level attention, and sentence-level attention. In our approach, we only considered the sentence-level attention. The latter follows the same spirit as in (Zichao Yang et al. 2016). In essence, the attention mechanism assigns a hidden representation u_i to the annotation h_i (generated from either BiLSTM or BiGRU encoder) of the i^{th} sentence and the associated weight α_i is constructed as the normalized version (using softmax function) of the similarity between the i^{th} hidden representation u_i and some sentence-level context vector U_s as follows:

$$u_i = \tanh(W_s h_i + b_s) \quad (3.7)$$

$$\alpha_i = \frac{\exp(u_i^T U_s)}{\sum_t \exp(u_t^T U_s)} \quad (3.8)$$

$$v = \sum_t \alpha_t h_t \quad (3.9)$$

where v is the document vector that summarizes all the information of sentences in a document. W_s and b_s are the weight and the bias from the attention layer, respectively. The sentence-level context vector U_s is seen as a high-level representation of a fixed query "what is the informative sentence" over the whole set of sentences. We randomly initialize the context vector U_s , and jointly learned during the training process.

In (9), a weighted sum of the word annotations based on the learned weights is computed as a representative for all sentences of the corpus. See Figure 3.2 for a detailed graphical representation of BiGRU additive-attention model (a similar representation applied to BiLSTM attention model).

Classification module We pass the output vector to a non-linear activation function to assign to each vector the corresponding class. This is performed using a simple sigmoid layer with 2 neurons (one for positive sentiment and another one for negative sentiment). The output

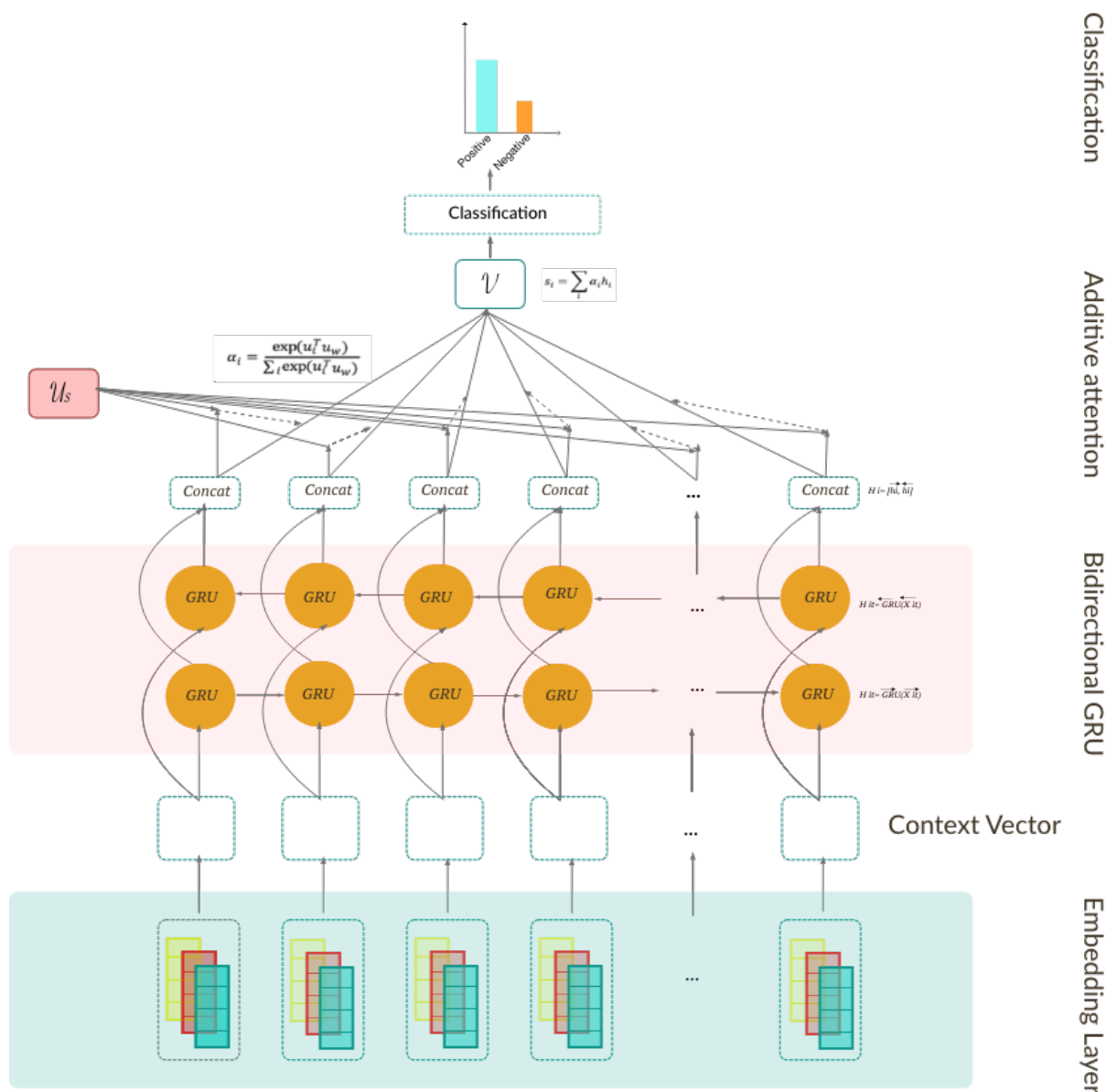


Figure 3.2: The proposed BiGRU additive-attention model

of this layer is a probability distribution that sums up to 1, so that the neuron having the higher probability value corresponds to the label of the sentence.

3.1.3 Experimentation

Datasets

Deep learning models usually perform well on large-scale datasets (Goodfellow et al. 2016), but collecting and annotating such data could be challenging and time-consuming. Therefore we used some of existing Arabic datasets for sentiment analysis. The data selection criteria are based on the data size (large enough to make the reasoning sound) and popularity within the

Arabic NLP research community. For this purpose, we used three large-scale Arabic sentiment analysis datasets whose reviews were written in MSA (Modern Standard Arabic) and Arabic dialects.

LABR Dataset The large-scale Arabic Book Reviews Dataset (LABR) (Aly et al. 2013) compiles a list of Arabic book reviews and has 63257 reviews taken from the Goodread book’s website. In this work, we used the binary annotated version (positive or negative labels) of the dataset which contains 51056 reviews. The annotations were made based on the reviewers’ ratings, such that 4-5 stars reviews were labeled as positive, 1-2 star reviews were labeled as negative, and, finally, reviews of 3 stars were dropped.

HARD Dataset The *Hotel Arabic-Reviews Dataset* (Elnagar, Y. Khalifa, et al. 2018) is a large-scale Arabic dataset that is widely used in the Arabic sentiment analysis research community. The dataset contains 105698 Arabic reviews written in both MSA and Arabic dialects on hotels collected from Booking.com. In this study, we used the balanced binary version of the HARD dataset, which groups 52,849 for each positive and negative review.

BRAD dataset (Elnagar and Einea 2016) is a large-scale annotated dataset of almost 510,600 book records in Arabic language where each record corresponds to a single review and the reviewer’s rating on a scale of 1 to 5 stars. We used the balanced version of BRAD, which contains 156K positive and negative reviews.

Table 3 summarizes the main statistics of the aforementioned three datasets.

Table 3.1: The size of different datasets

| Dataset | Nb. of Positive posts | Nb. of Negative posts |
|-------------|-----------------------|-----------------------|
| LABR | 42832 | 8224 |
| HARD | 52,849 | 52,849 |
| BRAD | 78380 | 78380 |

Baselines

To validate the performance of our model, we first select a set of commonly employed architectures, which would form our baseline, as alternatives to attention-mechanism-based architecture sentiment analysis. The choice of these models is motivated by the desire to seek the contribution of the attention mechanism module alone. Therefore, we selected deep learning architectures

with an encoding layer and a fully connected layer that perform the classification task, but without an attention mechanism module. This consists of the following:

- Bidirectional RNN. More specifically, we have chosen two models (without the attention layer): the BiGRU and the BiLSTM. Both models were trained with the same hyperparameters to ensure a fair comparison.
- We trained these models on different embedding configurations using the Learnable embedding extracted from each dataset and we compared the result to the performance (with regard to accuracy and training time) of the FastText pretrained embeddings for the Arabic language.
- Across all experiments, we used five performance metrics: accuracy, micro-Accuracy, micro-Precision, micro-F1 measures and epoch training time:

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

$$Micro - Precision = \frac{TP_1 + TP_2 + \dots + TP_n}{TP_1 + TP_2 + \dots + TP_n + FP_1 + FP_2 + \dots + FP_n}$$

$$Micro - F_1 = \frac{2 \times Precision_{Mirco} \times Recall_{Mirco}}{Precision_{Mirco} + Recall_{Mirco}}$$

where TP refers to the number of positive sentences (input sequences whose sentiment class is positive) and was predicted as positive as well by the model.

TN is the number of negative sequences that were classified by the model as negative as well.

FP is the number of negative sequences that are wrongly classified as positive .

FN is the number of positive sequences that are wrongly miss-classified as negative sequences.

The higher the precision value, the more accurate the prediction of the positive class. Similarly, a high recall value indicates that a high number of sentences from the same class are labeled to their exact class. While F1-measure is a weighted average of Precision and Recall, summarizing the ratio of the correctly classified sentences regardless of their class. Epoch training time corresponds to the time that the model took to train and validate on a single epoch (single pass over the dataset).

Hyperparameters tuning

Deep learning algorithms, including BiLSTM, BiGRU and Attention mechanism have various hyperparameters that may control and affect the training behavior, memory allocation, execution time, and even the models' performance. A common practice is to perform a grid search on a small finite set of parameters to select optimal values (Goodfellow et al. 2016). Typically, the hyperparameter search ensures that we do not make an opportunistic selection. Therefore, we performed a grid search that trains the associated model (s) for every joint specification of hyperparameter values on the four model configurations (BiLSTM and BiGRU (baseline), BiLSTM-Attention, BiGRU-Attention) with each of the three datasets. The hyperparameter configuration that achieves the minimal validation error was then chosen as corresponding to the best hyperparameters set, see Table 4.

Table 3.2: Summary of hyperparameter (HP) selection (the bold item is the chosen HP)

| Hyperparameters | Value |
|---------------------------|--------------------------------|
| Dropout_rate | 0.3 — 0.2 — 0.4 |
| Learning rate | 0.0001 — 0.01— 0.001 |
| Optimization algorithm | Rmsprop— Adam — Adagrad |
| EarlyStopping | 10 epochs |
| Batch size | 128 — 64— 250 |
| Number of Recurrent cells | 250 — 128 |
| Number of epochs | 25 |

3.1.4 Results

Training and Accuracy on Sentiment Dataset

After training our baselines on the best hyperparameter set highlighted in Table 4, we compared the outcomes of each baseline model using learned embedding and FastText embedding on each dataset. The results in terms of Accuracy, Precision, and F1-score of the three sentiment datasets are summarized in Tables 3.5, 3.3 for LABR and HARD dataset and Table 3.4 for the BRAD dataset. Initially, we looked at the training performance of the two baseline models BiGRU and BiLSTM, where it showed, possibly because of their simple architecture, that BiGRU tends to outperform BiLSTM as illustrated in Table 3.5. Besides, the use of pretrained embeddings made the training of the gated networks smoother and much faster. Therefore, we trained our additive-attention model on the BiGRU configuration and we regularly monitored the model’s performance.

Table 3.3: Sentiment Analysis results for the HARD dataset

| HARD dataset | | | | | | |
|--------------|-----------------|--------------|----------|-----------------|--------------|-------------------|
| Model | Embeddings | Acc(%) | MA-F1(%) | MA-Precision(%) | MA-recall(%) | Training Time (s) |
| BiGRU | Learned Embed. | 94.80 | 95.10 | 96.30 | 96.31 | 449 |
| | FastText Embed. | 94.84 | 95.59 | 96.21 | 96.38 | 175 |
| BiLSTM | Learned Embed. | 94.11 | 94.92 | 96.31 | 96.38 | 456 |
| | FastText Embed. | 94.77 | 95.32 | 96.38 | 96.38 | 175 |
| Our model | Learned Embed. | 95.79 | 95.91 | 96.88 | 96.1 | 436 |
| | FastText Embed. | 96.29 | 96.28 | 97.03 | 96.14 | 169 |

Table 3.4: Sentiment analysis results for the BRAD dataset

| BRAD dataset | | | | | | |
|--------------|------------------|--------------|--------------|-----------------|--------------|-------------------|
| Model | Embeddings | Acc(%) | MA-F1(%) | MA-Precision(%) | MA-recall(%) | Training Time (s) |
| Bigru | Learnable embed. | 93.23 | 91.12 | 94.12 | 94.32 | 372 |
| | FastText | 93.23 | 91.10 | 94.19 | 94.25 | 163 |
| BiLSTM | FastText | 92.88 | 92.17 | 94.10 | 93.87 | 165 |
| | Learnable embed. | 93.05 | 92.08 | 97.13 | 93.69 | 374 |
| Our model | FastText | 95.10 | 95.94 | 96.73 | 96.96 | 375 |
| | Learnable embed. | 95.65 | 96.15 | 97.21 | 97.10 | 165 |

Table 3.5: Sentiment analysis results on LABR dataset

| LABR dataset | | | | | | |
|--------------|-----------------|--------------|--------------|-----------------|--------------|-------------------|
| Model | Embeddings | Acc(%) | MA-F1(%) | MA-Precision(%) | MA-Recall(%) | Training Time (s) |
| BiGRU | Learned Embed. | 83.22 | 90.62 | 83.30 | 96.16 | 261 |
| | FastText Embed. | 83.04 | 89.1 | 86.03 | 96.38 | 25 |
| BiLSTM | Learned Embed. | 84.2 | 89.10 | 89.90 | 95.96 | 265 |
| | FastText Embed. | 85.83 | 90.10 | 90.03 | 96.10 | 26 |
| Our model | Learned Embed. | 95.71 | 96.3 | 96.81 | 96.88 | 258 |
| | FastText Embed. | 95.73 | 97.13 | 97.41 | 97.10 | 23 |

One notices for instance that our approach achieves 14.9%, 1.4%, and 2.5% improvement over

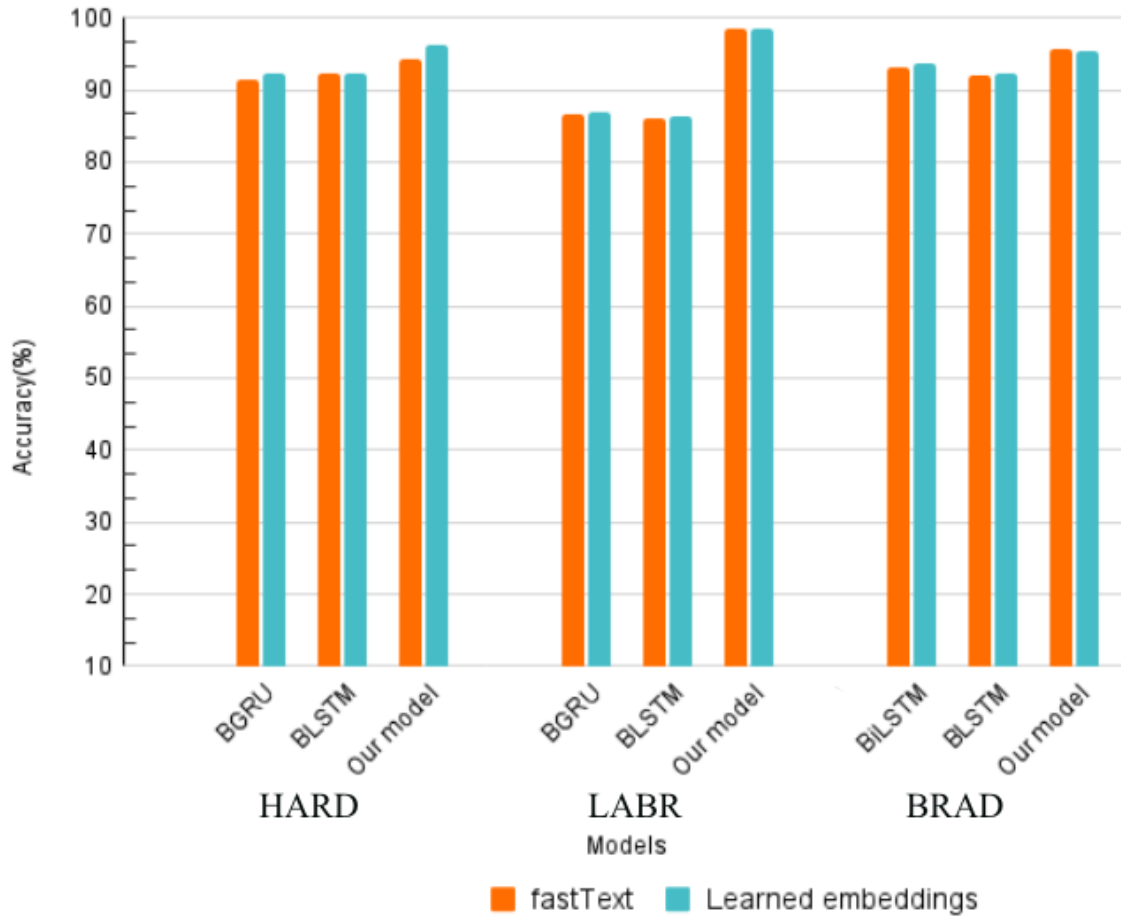


Figure 3.3: Classification report of our baselines compared to our model.

(best) baseline in the case of LABR, HARD, and BRAD dataset, respectively. The results in terms of Micro-Precision, Micro-Recall, and Micro-F1 scores reported in Table 3.4, 3.3 also indicated that the same trend of the superiority of our developed model is still noticeable.

Besides, to compare our results with some state-of-the-art results reported in the analysis of the three datasets, Table 3.5 summarizes the comparison in terms of the accuracy of our model and previous works using the same dataset.

The results highlight that the developed model outperforms the state-of-the-art sometimes by a large margin (e.g., 12.5% in case of BRAD dataset). In the same table, we reported the result when using a non-deep learning model, where results using support vector machine classifier have been reported. The choice of this classifier is motivated by the fact that it outperforms all machine learning classifier models (non-deep learning models) implemented in Orange Data

Mining library ². As it can be noticed, our model also outperforms SVM by a large margin.

Table 3.6: Comparison of sentiment analysis results on different datasets

| Dataset | Previous works | Result (Acc) | Our Model (Acc) |
|---------|---|--------------|-----------------|
| HARD | SVM (Elnagar, Y. Khalifa, et al. 2018) | 92.7% | |
| | LR (Elnagar, Y. Khalifa, et al. 2018) | 93.5% | 96.29% |
| | ULMFiT (ElJundi et al. 2019) | 95.7% | |
| | AraBERT (Antoun et al. 2020) | 96.1% | |
| | mBERT (Devlin et al. 2018) | 95.7% | |
| LABR | SVM (ElSahar et al. 2015) | 78.3% | |
| | AraBERT (Antoun et al. 2020) | 89.6% | |
| | Multi-chan CNN (A. Dahou et al. 2019) | 87.5% | 95.65% |
| | SRU-Attention (Al-Dabet and Tedmori 2019) | 95.1% | |
| | SVM (Altowayan et al. 2016) | 81.27% | |
| BRAD | SVM (Elnagar and Einea 2016) | 85% | |
| | LR (Elnagar and Einea 2016) | 84.4% | 95.73% |

3.1.5 Statistical evaluation

We would like to find out whether our model superiority in the three datasets (LABR, HARD and BRAD) is statistically significant. For this purpose, we run a statistical test (t-test) to check that the population of results corresponding to our model is different from the population of results corresponding to the second-ranked one. The null hypothesis of the test assumes that the means of the two populations are equal. Therefore, rejecting the null hypothesis indicates that the results of the two close populations are different, which means that the order between the two results is statistically significant. Besides, as we are only interested in the significance of the difference regardless its direction, a one-tailed t-test was employed. To create the population, we use various randomization of the training samples to create slightly distinct realizations of our model and alternative model. Table 3.7 summarizes the result of the t-test. Using a 95% confidence interval, it is easy to see that in all cases, the null hypothesis is rejected $p < 0.0001$, which indicates that the initial statement of the superiority of our model is statistically significant as well.

Table 3.7: Statistical test on the performance of the proposed model and the BiGRU model.

| Dataset | HARD | | LABR | | BRAD | |
|-------------|--------|----------|--------|---------|--------|----------|
| | T-test | Pvalue | T-test | Pvalue | T-test | Pvalue |
| Test Result | 75.01 | < 0.0001 | 89.41 | <0.0001 | 85.01 | < 0.0001 |

²<https://orange3.readthedocs.io/projects/orange-data-mining-library/en/latest/index.html>

3.1.6 Discussions and Implications

Significance of the results

- Unlike traditional methods presented by (Elnagar and Einea [2016](#)) and (Elsahar et al. [2015](#)) that investigated the Arabic sentiment analysis problem with standard TF-IDF and n-grams features, our approach proposed a new deep learning approach that uses a combination of FastText and Learned embeddings in an attention layer architecture to handle the ASA problem. The results showed a clear outperformance compared to the state-of-the-art results presented in the reference papers of the employed datasets, which testify to the feasibility and the attractiveness of the proposed method in terms of accuracy.
- The comparison between the bidirectional GRU and BiLSTM showed that the former model is less parametrized, which resulted in a smaller training time compared to that of the BiLSTM model. This result is in agreement with other findings in the literature as well, see, e.g., (Zichao Yang et al. [2016](#)) and the whole special issue of the underlined IEEE Access journal.
- The experiment results revealed that the attention-based model gave significantly better results than other recurrent nets-based models without attention mechanisms.
- The implemented attention mechanism offers the possibility to zoom on specific wordings or constructs according to their contribution to the sentiment polarity as per their conceptual model. This can lie bare foundation for an enhanced visualization toolkit providing some explanation to the findings.
- The use of pretrained embeddings decreased the training time because there is no learning in the embedding layer. This is highlighted during the comparison between FastText embeddings and Learned Embeddings when the gain in terms of learning time can be of the order of two to ten times (as in the case of LABR dataset). Nevertheless, this does not mean that the fast learning model yields better accuracy results.
- As pointed out in Table [3.6](#), our developed model outperformed all previous works on the three datasets, including the state-of-the-art models such as AraBERT (Antoun et al. [2020](#)). Without questioning the theoretical foundation and sound empirical foundation of AraBERT, we believe that the fine-tuning mechanism of the weights brought by the

attention mechanism together with the initial setting up strategy played a central role in enhancing the result of our approach.

- The conducted experiments strongly suggest that incorporating the attention mechanism with BiGRU can boost the training time of neural network models. The proper mechanism that allows them to pay attention to only parts of sequences and extract meaningful information can effectively detect texts' sentiments.
- When incorporated with BiGRU /BiLSTM networks, the attention mechanism highlights an essential part of the sentiment analysis task on the entire sequence.
- It would have been interesting to see the performance of our model with respect to other related attention mechanism-based models mentioned in the related work section. Nevertheless, the absence of open-source implementation and a clear Arabic language focus renders the reproduction of such implementations quite difficult and possibly not faithful.

Research implications

- The developed attention mechanism architecture that can be accommodated with minimum changes to distinct NLP tasks offers a nice opportunity for the research community to handle several tasks simultaneously. This corresponds to a substantial shift from a single-task model, which substantially dominated the current practice in natural language processing and machine learning tasks.
- The finding that learnable embedding provides better results than large-scale FastText embeddings, may challenge the global trend and thought that such large-scale embeddings (e.g., FastText, Word2vec, Glove) are the preferred choice in the design of deep learning models.
- The use of a simple voting scheme for choosing between BiLSTM and BiGRU layers in the developed model, although it is widely under-investigated in deep learning literature, should be taken with caution where more elaborated schemes can also be implemented. Indeed, this can be cast under the general framework of a combination of independent classifiers or dynamic classifier selection. In this respect, several elaborated metrics have been put forward to guide such selection or hybridization scheme, which includes individual classifier's accuracy, ranking, probabilistic based-measures, behavior-based measures,

and diversity-related measures, among others (P.R. Cavalin [2013](#)). Such measures can be defined on either the whole feature space or according to some predefined partition. The input space can also be partitioned according to other rational criteria. So far, our model uses the whole input post to dynamically choose between BiLSTM and BiGRU. Nevertheless, other researchers have considered a partition that involves linguistic constructs, attempting to look at whether, e.g., BiLSTM is better when the post contains negation, and/or, adjective/adverb, location-named entity, and question mark, among others.

- Alternative to dynamic classification systems, the voting strategy can also be cast into the framework of meta-learning theory (Vilalta et al. [2002](#)) where individual classifiers are considered as separate learners and the outcome of the voting scheme as the meta-learning result. Ultimately linked to this viewpoint are the meta-features that can be setup initially to guide the performance of the individual classifiers. Especially, it is worth identifying whether one can distinguish individual features that yield better performance for BiGRU and those for BiLSTM. Other potential extensions consist in creating several versions of BiLSTM and BiGRU by utilizing several learning algorithms on the same dataset and use a stacking framework for a weighted combination yielding a more elaborated voting scheme.
- The use of the attention mechanism through its weighting scheme of the various rule offers a nice setting to contribute to the explainable AI, a field that is expected to substantially grow in the near future as per EU Artificial Intelligence new policy for instance [3](#).

Recurrent neural networks have brought a breakthrough into the NLP field, where most state-of-the-art and recent studies on sentiment analysis used RNN and deep learning architectures. The use of attention layers has also shown a growing interest to enhance the explainability and interpretability of the results. This study contributes to this field with a focus on Arabic sentiment analysis where a new attention mechanism-based deep learning approach has been put forward and tested using publicly available datasets.

The proposed model assumes two types of embeddings FastText and Learnable embeddings. Besides, Bidirectional GRU /LSTM, which reads the individual sentence input from left to right and vice versa to capture the contextual information was employed. On the other hand, for learning long and short dependencies, we used two sets of recurrent gated networks; namely,

³<https://digital-strategy.ec.europa.eu/en/policies/strategy-artificial-intelligence>

BiLSTM and BiGRU where an ad-hoc hybridization strategy has been devised through a simple voting strategy that makes use of training quality and performance of individual BiLSTM and BiGRU models on individual input.

The study also showcases the performance of the various embedding strategies. A comparison between baseline models revealed the simplicity of GRU cell concept compared to LSTM, which yields a better training performance.

Our model has been tested on three wide-scale employed Arabic sentiment datasets: LABR, HARD, and BRAD. The testing demonstrated that our model outperforms both baseline models and state of art models reported in the original references of these datasets. Furthermore, to demonstrate the generalization capabilities of our model, the performance with respect to news categorization, offensive speech detection, and Russian sentiment analysis tasks has been carried out. Remarkable results have been noticed.

Perspective work In terms of perspective work, we believe that there is a room for further fine-tuning the parameters of the model to enhance its performance and provide a more theoretical foundation for the hybridization mechanism. For instance, the recent advances in the application of meta-heuristic for parameter optimization, e.g., Laith, Dalia, et al. (2021), Laith and D. Ali (2021), and Laith, D. Ali, et al. (2021), can provide insights to a better fine-tuning of the deep-learning parameters of our attention-mechanism based architecture, or optimizing postprocessing stages as in (Abualigah et al. 2016; Abualigah 2018). Besides, attention weights can further be explored to guide interpretability and explainability mechanism. This can enhance the development of an appropriate user interface.

The results in the case of the News Classification dataset revealed that our model is outperformed by several other state-of-the-art and deep-learning model architectures. This probably highlights the importance of the preprocessing stage, which has not been detailed in related studies, and therefore, kept minimal in our cases. Therefore, there is room for improvement in this regard. This possibly also shows the potential vulnerability of the architecture when trying to extend from binary class to multi-class cases.

3.1.7 Conclusion

Recurrent neural networks have brought a breakthrough into the NLP field, where most state-of-the-art and recent studies on sentiment analysis used RNN and deep learning architectures. The use of attention layers has also shown a growing interest to enhance the explainability and interpretability of the results. This study contributes to this field with a focus on Arabic sentiment analysis where a new attention mechanism-based deep learning approach has been put forward and tested using publicly available datasets.

The proposed model assumes two types of embeddings FastText and Learnable embeddings. Besides, Bidirectional GRU /LSTM, which reads the individual sentence input from left to right and vice versa to capture the contextual information was employed. On the other hand, for learning long and short dependencies, we used two sets of recurrent gated networks; namely, BiLSTM and BiGRU where an ad-hoc hybridization strategy has been devised through a simple voting strategy that makes use of training quality and performance of individual BiLSTM and BiGRU models on individual input.

Our model has been tested on three wide-scale employed Arabic sentiment datasets: LABR, HARD, and BRAD. The testing demonstrated that our model outperforms both baseline models and state of art models reported in the original references of these datasets. Furthermore, to demonstrate the generalization capabilities of our model, the performance with respect to news categorization, offensive speech detection, and Russian sentiment analysis tasks has been carried out. Remarkable results have been noticed.

3.2 Speech Emotion Recognition for Algerian dialect

Speech Emotion Recognition (SER) is a rapidly growing area of research, with a focus on using deep learning to develop accurate models to automatically recognize the emotion of the speaker. This chapter expands the analysis of emotions in Arabic text and focuses specifically on emotion recognition from speech signals in the Algerian dialect. We first begin by describing the methodology of collecting a realistic speech corpus from Algerian TV shows, which serves as the foundation for our subsequent work. Furthermore, we train classical ML-based models and neural network-based models to efficiently recognize four emotions.

Through our research, we aim to contribute to the development of more accurate and effective SER models for the Algerian dialect. This work has important implications for a variety of applications, including virtual assistants, speech therapy, and affective computing. Overall, this chapter provides a comprehensive overview of our approach to emotion recognition from speech signals in the Algerian dialect, and our efforts to advance this field through the use of deep learning techniques.

3.2.1 Introduction

SER is an emerging area of research in NLP that focuses on recognizing and analyzing emotions in human speech. Particularly, SER aims to accurately identify and classify emotions conveyed through speech, including happiness, sadness, anger, fear, and other affective states.

SER has become increasingly important in various applications, including human-computer interaction, virtual assistants, mental health diagnosis and treatment, and social robotics. For example, SER can be used to analyze and respond appropriately to the emotional states of users interacting with a virtual assistant (Chatterjee et al. 2021) or to monitor and assess the emotional state of patients during therapy sessions (Egger et al. 2019).

SER is a challenging task due to the complexity and variability of human emotions and speech (R. A. Khalil et al. 2019). Emotions can be expressed in various ways, including pitch, intonation, tone, rhythm, and speech rate. Moreover, emotions can be influenced by cultural, social, and contextual factors, making it difficult to develop accurate and reliable SER models.

Upon conducting a thorough investigation into relevant literature concerning the Arabic lan-

guage and its various dialects, it became evident that only a sparse number of studies have been conducted. This phenomenon can be largely attributed to the scarcity of available data and lack of interest from the research community as well as from enterprises in the Arab region in the development of intelligent products and assistants that use speech technology. Consequently, there is a pressing demand for research to be carried out on the Arabic language, more specifically the Algerian dialect, as this dialect is underrepresented in academic discourse. Deep learning techniques have demonstrated significant benefits in the field of speech analysis and speech emotion recognition. Thanks to the specific nature of recent deep learning models such as CNN (LeCun et al. [1989]) and LSTM (Hochreiter et al. [1996]) in learning intricate patterns and representations from raw speech data, has enabled them to capture subtle acoustic features and temporal dependencies, which are crucial for accurate speech analysis and emotion recognition. Additionally, deep learning models have the flexibility to learn end-to-end representations directly from speech data, eliminating the need for handcrafted feature engineering (Fayek et al. [2017]). This feature extraction capability alleviates the manual effort required in traditional approaches and enables the models to learn more representative and task-specific features, thereby improving the accuracy of speech analysis and emotion recognition systems.

The following contributions have been made toward advancing SER applications in the Arabic language:

- **Dataset Development:** One fundamental contribution has been the creation and expansion of the Arabic speech emotion dataset. This dataset comprises 1202 high-quality WAV files annotated samples from Algerian speakers expressing various emotions. The availability of such datasets has facilitated the training and evaluation of SER models, enabling researchers to develop more robust and contextually relevant systems. The records were carefully selected to capture a diverse range of natural speech patterns and emotions.
- The annotation of the speech signals with respect to the different emotions expressed, namely happy, angry, neutral, and sad. This process involved the subjective assessment of the emotional content of each recording by a group of human annotators, which was then compiled into a comprehensive emotion-label dataset.
- Efficient data processing pipeline, which involved various tasks such as audio normalization, noise reduction, and feature extraction, in order to prepare the dataset for the subsequent stages of analysis and modeling.

- The development of six classification models using both machine learning and deep learning approaches, each tailored to accurately predict the emotional state of a given speech signal in the Algerian dialect. These models included popular techniques such as SVM, LSTM, and CNNs, as well as novel hybrid models combining multiple methods.
- The comparison of the proposed models against various baselines, including simple rule-based models and SOTA approaches from related studies, in order to evaluate the effectiveness and generalizability of the developed models.

3.2.2 Methodology

This section outlines the research methodology employed to tackle the task of devising efficient models for SER in the Algerian dialect. The methodology encompasses two primary phases: (i) the creation and annotation of a novel speech corpus specific to the Algerian dialect, and (ii) the development and training of diverse classification models for effectively recognizing emotions from speech data. By detailing these methodological steps, this section provides a comprehensive understanding of the systematic approach undertaken to achieve the desired objectives in the realm of SER for the Algerian dialect.

ASER Dataset

In our research, we put forth a proposal to develop a comprehensive database of Algerian dialect Speech Emotion Recognition (ASER). The proposed database comprises 1202 high-quality WAV files, which were carefully selected to capture the nuances and subtleties of the Algerian dialect. The duration of the recordings varied from one second to seven seconds, providing a diverse range of natural speech patterns and emotional expressions. The collection process and pipeline of ASER is presented in Fig. [3.4](#).

The primary aim of ASER dataset is to serve as a valuable resource for researchers and engineers in the field of natural language processing, particularly for those focusing on the Algerian dialect. The database can be utilized for a wide range of tasks, such as speech recognition, emotion detection, and sentiment analysis. Also, the creation of ASER dataset is an important step toward advancing the field of speech processing in the Algerian dialect, which has been largely understudied and underrepresented in academic discourse. By providing a comprehensive and diverse collection of high-quality speech recordings, this dataset will facilitate research into var-

ious aspects of the Algerian dialect, leading to deeper insights and a better understanding of this language variety.

ASER comprises speech recordings extracted from a variety of Algerian TV shows, including "Imarat lhadj Lakhdhar", "Soug nssa", and "Taht lmourakaba", noted as source definition in the pipeline Fig. 3.4. The speech recordings were carefully selected to capture a diverse range of natural speech patterns and emotional expressions, ensuring that the dataset is representative of the Algerian dialect in different contexts. Furthermore, the ASER dataset passed through the data filtering stage so it encloses recordings from a diverse set of speakers, including 21 males, 12 females, and 4 children, thereby ensuring a broad representation of age and gender. Thereafter, each speech recording in the dataset is annotated with one of four distinct emotions: neutral, angry, happy, or sad. To ensure accuracy, two native Algerian speakers with annotation expertise were involved in the labeling process. They listened to each recording and identified the primary emotional state conveyed. In instances where uncertainty arose, the two annotators collaboratively reviewed and discussed the annotations to reach a consensus on the appropriate label.

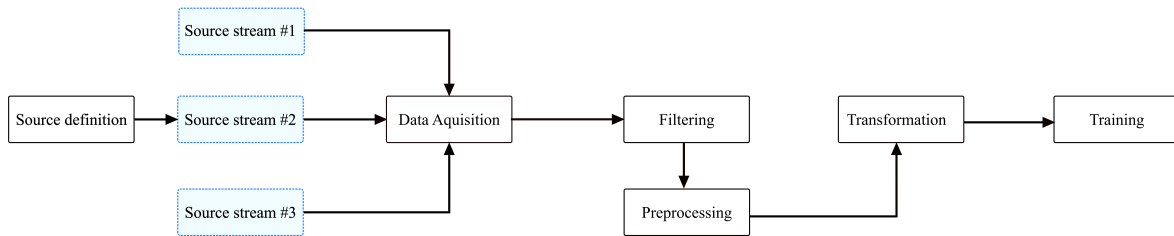


Figure 3.4: ASER Data collection pipeline.

Table 3.8 presents an overview of the label distribution of ASER, highlighting the number of speech recordings available for each emotion and speaker category.

Data Preparation

In speech analysis, data preprocessing plays a crucial role in ensuring that the input signals are clean and ready for analysis. In our methodology, we have applied a series of preprocessing techniques to our collected Algerian dialect dataset to prepare it for the subsequent stages of our analysis.

The primary objective of preprocessing is to eliminate any unwanted noise or distortions that

may interfere with the analysis of the speech signals. To achieve this, we have employed various techniques, including noise reduction, normalization, and feature extraction.

Normalization is another important preprocessing technique that aims to standardize the amplitude of speech signals. This technique is essential because the recorded speech signals may have different amplitudes, which could impact the performance of the machine learning models. Therefore, we have applied normalization techniques to ensure that the signals have a consistent amplitude, making them more reliable for further analysis.

Feature extraction is the process of selecting and transforming the most relevant information from the raw speech signals. We have employed a range of feature extraction techniques to extract features such as mel-frequency cepstral coefficients (MFCCs) and prosodic features from our speech dataset. These features are widely used in speech recognition and emotion detection tasks, and they provide useful information about the speech signals that can be used to train and evaluate machine learning models.

Table 3.8: Overview of class distribution in the ASER dataset.

| Emotion | Nb. of records |
|------------------|-----------------------|
| Happy | 123 |
| Neutral | 473 |
| Sad | 253 |
| Angry | 353 |
| Total utterances | 1202 |

Data augmentation Data augmentation techniques are commonly used to increase the size of the training dataset, thereby improving the performance of machine learning models. In our study, we have employed two data augmentation techniques to our collected Algerian dialect dataset: pitch tuning and white noise adding.

- **Pitch tuning:** is a technique that involves altering the pitch of the speech signals by shifting the frequency of the audio signals up or down. This technique can be used to simulate the natural variation of the human voice, which can improve the robustness of the ML models. We applied pitch tuning to our dataset by randomly shifting the pitch of each audio file up or down by a certain percentage. This technique allowed us to generate new samples of our data, thereby increasing the size of our training dataset.
- **White noise adding:** is another data augmentation technique that involves adding a small amount of random noise to the audio signals. This technique can help to simulate

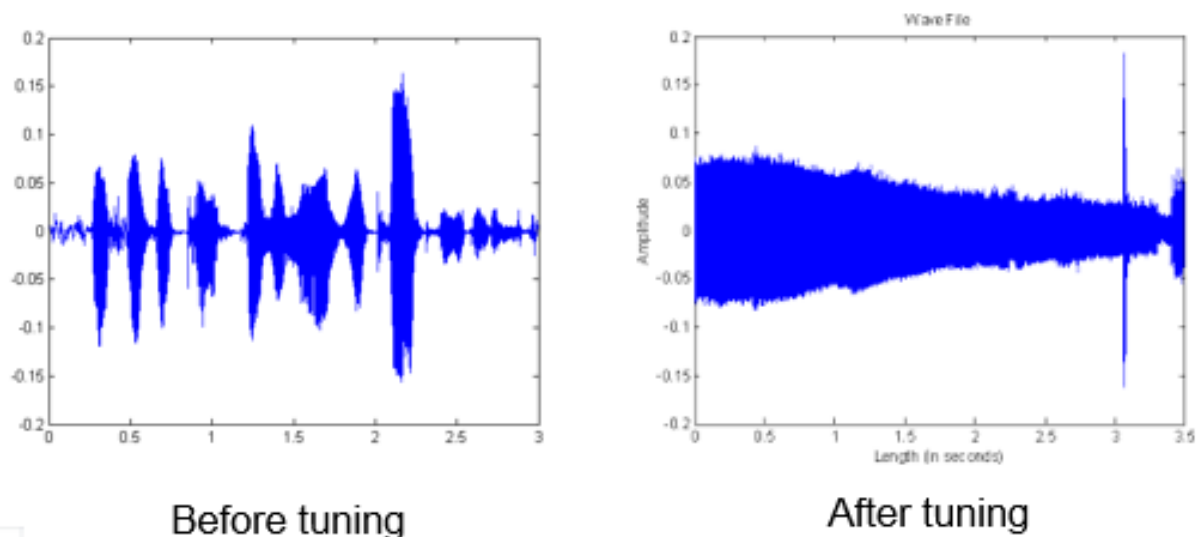


Figure 3.5: Raw WAV file before and after pitch tuning.

real-world conditions, where background noise is often present in speech signals. We applied white noise adding to our dataset by adding a small amount of white noise to each audio file, as presented in Fig. 3.6.

The white noise signal at each time point t is expressed as the product of an amplitude factor, denoted as A , and a random variable $\epsilon(t)$. The random variable $\epsilon(t)$ follows a probability distribution with zero mean and unit variance. Typically, it is assumed to conform to a Gaussian distribution, representing a series of independent and identically distributed random values. Multiplying each of these values by the amplitude factor A scales the white noise signal according to the desired level of noise to be added.

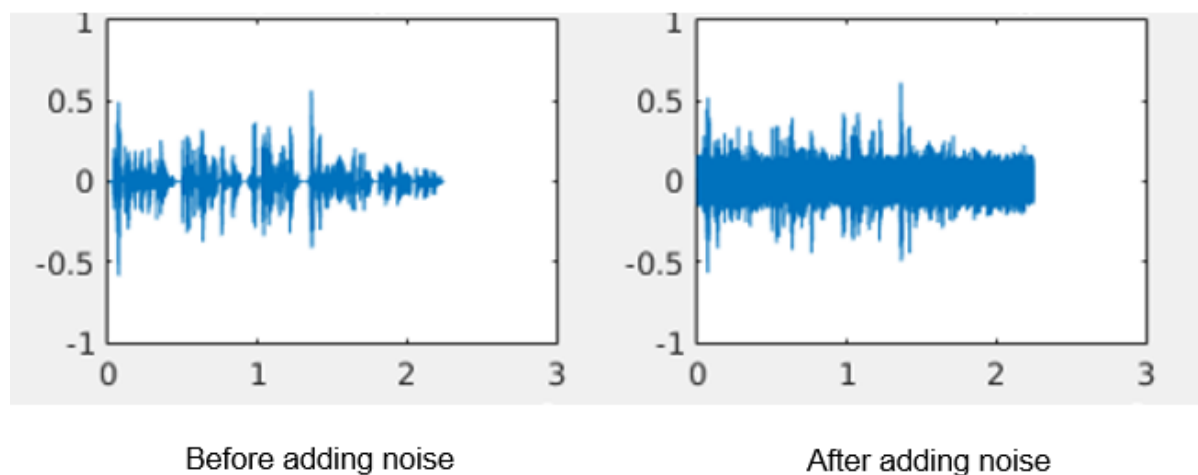


Figure 3.6: Raw WAV file before and after adding white noise.

3.2.3 Baseline models

One primary objective of this chapter is to conduct a comprehensive performance comparison between classical ML models and deep neural network models regarding feature extraction and SER tasks. To achieve this, we utilized the ASER dataset and trained various classical ML models, namely the decision tree classifier, random forests, and support vector machine (SVM). Subsequently, we compared their performance against three novel neural network models proposed in this study, which are presented as follows:

The BLSTM-CNN, encompasses a total of eight convolution layers, two max-pooling layers, one dense layer, two dropout layers (Srivastava et al. 2014), and two batch normalization layers (Ioffe et al. 2015). The dropout rate is set to 0.25, and a batch size of 16 is employed during training. The learning rate is set at 0.0001, and the model undergoes training for 500 epochs. This architecture contains 1.4 million learnable parameters, and each epoch requires approximately 2 seconds for completion.

3.2.4 Proposed DeepEmoNet model

In our study, we present a new model that is presented in Fig. 3.7, and Fig. 3.8 that combines two distinct encoders: an LSTM-Encoder and CNN-Encoder. Combining effective feature representation from the two encoders boosts the feature extraction and thus the classification performance. The architecture of our proposed model comprises a set of LSTM layers, convolutional layers, and dense layers. The first layer is an LSTM layer with 259 neurons, which corresponds to the shape of the input signal, and 512 output neurons. The activations from this layer are then passed to the next layer.

The second LSTM layer is similar to the first one, with the output shape set to 256. The third, fourth, and fifth layers are convolutional layers, each with 256 filters of size 8 x 8 followed by the Rectified Linear Unit (ReLU) activation function. The output of the convolutional layers is flattened and fed into a fully connected layer, followed by a softmax function at the end of the model.

3.2.5 Experiments and results

In our research project, we conducted a comprehensive evaluation of our collected Algerian dialect dataset by conducting eleven experiments, each with a different combination of model

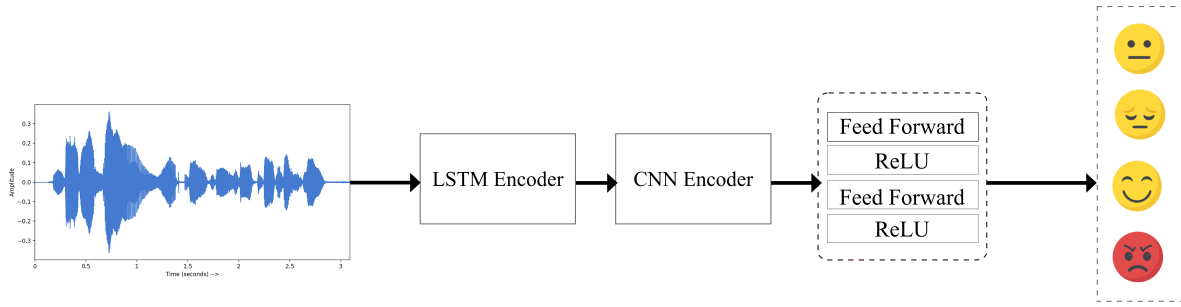


Figure 3.7: Generic graph for the proposed DeepEmoNet network for SER.

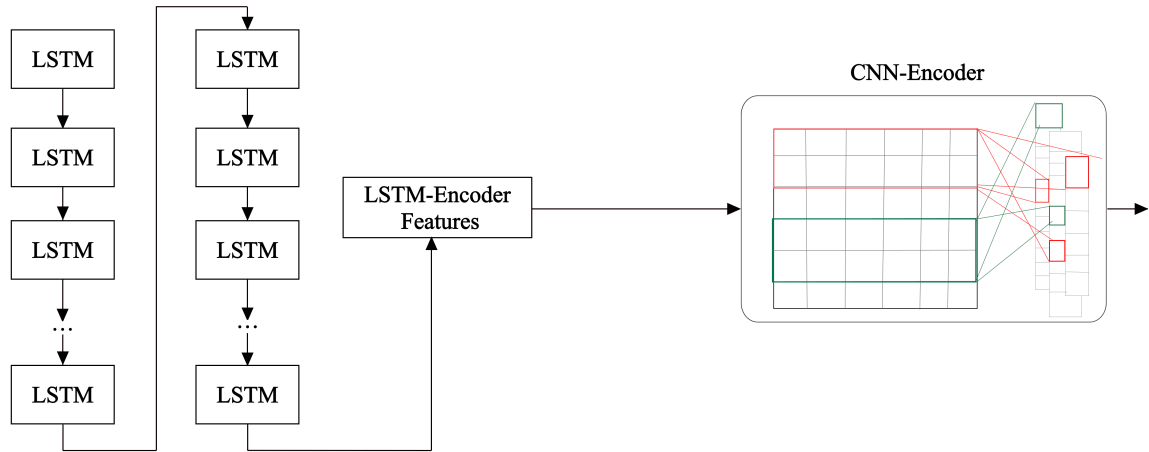


Figure 3.8: LSTM-CNN encoders for the proposed DeepEmoNet model.

architectures and hyperparameters. The aim of this evaluation was to identify the most effective model for accurately recognizing emotions in Algerian dialect speech signals.

To analyze the results obtained from each experiment, we compared the performance of each model based on several factors. Firstly, we analyzed the overall accuracy of each model on the test ASER dataset to determine which models were most effective at accurately recognizing emotions in Algerian dialect speech signals.

To evaluate the performance of our proposed hybrid model, we plotted the learning curve, as shown in Fig. 3.9. The learning curve was calculated based on the categorical cross-entropy loss, which is the metric used to optimize the parameters of the model. The results indicate that our model is generalizing well on the validation data, as there is no gap between the training and validation curves for both accuracy and test.

In addition, we also analyzed the generalization aspect of each model, by examining whether the model was prone to overfitting. Overfitting occurs when a model is too complex and has memorized the training data, resulting in poor performance on new, unseen data. To avoid

overfitting, we analyzed the validation accuracy of each model, which provides an estimate of how well the model generalizes to new data.

Table 3.9: Experimental results on ASER dataset.

| Model | Accuracy |
|-------------------|-----------------|
| Decision trees | 52% |
| SVM | 52% |
| Random Forests | 74% |
| CNN architecture | 77.53% |
| BLSTM-CNN | 92.93% |
| DeepEmoNet | 93.34% |

3.2.6 Discussion

The experimental results table displays the performance evaluation of various models based on their accuracy for the given task. The table presents the models tested and their corresponding accuracy scores.

Among the ML models, Decision Trees and SVM achieved an accuracy of 52%, indicating limited capability in accurately recognizing and classifying emotions from speech data. Random Forests performed slightly better with an accuracy of 74%, exhibiting a modest improvement over the other ML models. These ML models rely on handcrafted features and lack the ability to automatically learn intricate patterns and representations from the data.

In contrast, the deep learning models, namely the CNN architecture, BLSTM-CNN, and DeepEmoNet, significantly outperformed the ML models in terms of accuracy. The CNN architecture achieved an accuracy of 77.53%, showcasing the advantages of deep learning models in automatically extracting relevant features from raw data. The BLSTM-CNN model achieved a substantially higher accuracy of 92.93%, indicating the effectiveness of combining convolutional and recurrent neural networks for capturing temporal and spatial dependencies in speech data.

The DeepEmoNet model exhibited the highest accuracy of 93.34%, surpassing all other models in performance. DeepEmoNet leverages the power of deep neural networks with sophisticated architectures and a large number of learnable parameters. It can capture complex patterns and relationships in speech data, leading to enhanced emotion recognition capabilities.

The contrast in performance between deep learning models and traditional ML models can be attributed to several factors. Deep learning models have the ability to automatically learn

hierarchical representations from data, allowing them to uncover intricate patterns and relationships that may not be explicitly defined by handcrafted features. Deep learning models also have the advantage of scalability, as they can effectively handle large-scale datasets. Additionally, the ability of deep learning models to learn from raw data reduces the reliance on manual feature engineering, which can be time-consuming and limited in capturing the full complexity of speech data.

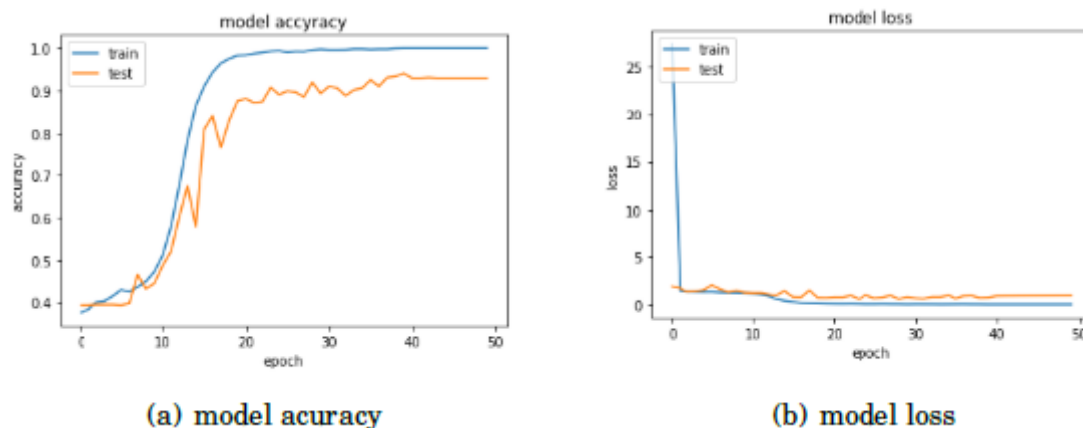


Figure 3.9: Loss and Accuracy Analysis of the Proposed DeepEmoNet Model.

The findings of our study have demonstrated that the utilization of deep neural networks can yield promising outcomes in emotion classification and can be widely applicable in speech emotion recognition tasks. Our proposed DeepEmoNet model has outperformed all other classifiers and approaches, exhibiting a high accuracy rate of 92.93%. The results have also indicated that the application of LSTM and CNN networks can produce remarkable results in speech emotion recognition, especially in our case study, which involves emotion detection from the Algerian dialect. Our approaches have demonstrated good results in terms of classification performance; however, we acknowledge that our models were not trained on a large dataset due to the limitations of the available resources.

3.2.7 Conclusion and future work

The present section presented a new case study with the purpose of constructing and analyzing an emotional speech corpus of the Algerian dialect. Additionally, this study seeks to propose a novel hybrid classification model for recognizing emotions from Algerian speech. In order to achieve the research objectives, a significant amount of effort was directed towards the collection and construction of a large annotated dataset comprising 1202 audio recordings that were categorized as happy, angry, neutral, or sad. Thereafter, numerous experiments were conducted using machine learning classification algorithms, as well as deep convolutional and recurrent neural networks.

The results of our research reveal that the implementation of the DeepEmoNet model holds great potential in SER tasks, and can be of substantial value across diverse domains. Nonetheless, it is acknowledged that the training of our models was confined by the limited resources available, thereby necessitating further exploration of the proposed model's efficacy with larger datasets.

In our future work, our objective is to broaden our research by incorporating state-of-the-art deep learning models such as Wave2Vec (Yuan et al. [2017](#)), SpeechBrain (Ravanelli et al. [2021](#)), HuBERT (Hsu et al. [2021](#)), and Whisper (Radford, J. W. Kim, et al. [2022](#)). By leveraging these advanced models, we aim to enhance the recognition of emotions in Algerian dialect speech, thereby improving the overall effectiveness of our study.

3.3 Inappropriate speech detection in Arabic text

In recent years, the rapid growth of social media platforms and online communication has led to an increased prevalence of inappropriate speech, which includes hate speech, offensive language, and cyberbullying. Detecting and mitigating such harmful content has become a critical concern for online communities, platform managers, and governments. The development and implementation of effective techniques for identifying inappropriate speech can contribute to the creation of a safe and respectful online environment, fostering healthy discourse and reducing the negative impact of harmful content on individuals and communities. In this section, we explore another field of Arabic speech analysis which is the analysis of social media posts and the detection of inappropriate speech using effective and advanced DL models in MSA and dialectal Arabic.

3.3.1 Introduction

The widespread of Internet use has made social media websites a ubiquitous presence in the lives of many individuals, particularly among adolescents. Recent research studies on the usage of social media services indicate that a substantial proportion of adolescents aged between 13 and 17 have utilized social media, with 51% reporting daily visits to these sites [\[4\]](#). On average, these adolescents spend nearly nine hours per day on social media, excluding time dedicated to academic pursuits. Despite the multitude of benefits associated with social media use, such as remaining connected with events of interest, assisting with academic duties, and providing access to relevant online communities and support, there are also potential negative impacts on mental health, such as harassment and exposure to online harm.

A recent survey reported that 50% of young people have been victims of online abuse [\[5\]](#). Furthermore, 13% of respondents reported experiencing cyberbullying at least once. Social media platforms can provide access to inappropriate content such as violent images or pornography.

A multicenter study that evaluated the emotional effects of various forms of bullying and cyberbullying found that 68.5% of adolescents experienced negative emotions such as anger, distress, and depression (Bottino et al. [2015](#)). Cyberbullying is a significant concern in Arab countries,

⁴https://www.aacap.org/AACAP/Families_and_Youth/Facts_for_Families/FFFGuide/Social-Media-and-Teens-100.aspx

⁵<https://www.common-sense-media.org/research/the-common-sense-census-media-use-by-tweens-and-teens-2019>

where it is becoming increasingly prevalent, especially among teenagers on social media. According to a survey, 20.9% of middle-school adolescents reported bullying in the UAE, 31.9% in Morocco, 33.6% in Lebanon, 39.1% in Oman, and 44.2% in Jordan (Kazarian et al. 2013).

The fundamental principle of social media is the provision of an unrestricted platform for the creation, sharing of information and ideas, and the expression of opinions and beliefs. However, the prevalence of cyberbullying has led to the regulation and moderation of these rights through policies implemented by various social media websites.

To mitigate the negative impact of cyberbullying, several social media websites have taken steps to address the issue. Facebook, for example, has a policy rationale that prohibits hate speech on its platform due to the intimidating and exclusionary environment it creates, and the potential for promoting real-world violence⁶. As a result, Facebook has implemented multiple tiers and restrictions on posts that target specific groups based on gender, ethnicity, or religious beliefs. Google has also taken a strong stance against hate speech by imposing strict policies and utilizing automatic services to detect and remove potentially violent content. Over 10,000 individuals have been tasked with reviewing and removing content that violates YouTube's guidelines, leading to the removal of over 17,000 channels, 100,000 videos, and 500 million comments⁷.

The determination of speech as hateful and offensive is contingent upon the context in which it is employed. For instance, members of a particular ethnic group may use terms that are typically considered insulting among themselves. In cases where the usage is consensual, the intention behind these terms is not malicious but rather a means of reclaiming words that were previously used to degrade members of their community⁸.

Parodies and sarcasm are linguistic phenomena that can convey both harmful intent and amusement, contingent upon the context and discourse in which they occur. For example, the statement "I will kill you" may be uttered in a humorous manner between individuals engaged in playful banter. However, comprehending the surrounding conversation becomes crucial in discerning whether the statement represents a genuine threat or not. Considering the increasing exposure of individuals to social media and online blogs, it has become imperative for researchers to assume significant responsibility in intervening and addressing these linguistic nuances. The

⁶[https://www.facebook.com/communitystandards/hate speech](https://www.facebook.com/communitystandards/hate%20speech)

⁷<https://youtube.googleblog.com/2019/09/the-four-rs-ofresponsibility-remove.html>

⁸<https://help.twitter.com/en/rules-and-policies/hateful-conduct-policy>

prevalence of such platforms necessitates a careful examination of the content and its intended meaning, particularly when it comes to potentially harmful or misleading messages. By delving into the contextual aspects and employing appropriate data analysis techniques, data scientists and researchers can play a pivotal role in identifying and mitigating the impact of such language use in online environments.

Most of the existing research in hate speech detection has predominantly focused on English data, leaving Arabic underrepresented due to multiple factors. Furthermore, a majority of the works have heavily relied on classical ML models and handcrafted features, resulting in limitations in word representation and linguistic nuances.

In our work, we aim to address these gaps by expanding the scope to encompass multiple Arabic dialects, including Libyan, Gulf, Algerian, Tunisian, as well as MSA. By incorporating a broader range of dialects, we aim to capture the rich linguistic diversity and cultural nuances present in Arabic speech.

Our contributions towards mitigating inappropriate Arabic speech in social media are multifaceted. Firstly, we propose a novel deep neural network model called ELSoA. ELSoA combines the power of word embeddings, LSTM networks, and soft attention mechanisms to effectively analyze and interpret Arabic speech data. This model enables us to delve into the intricate semantic and stylistic aspects of Arabic language usage, facilitating more accurate detection and classification of inappropriate content.

Secondly, we conduct a comprehensive evaluation of various deep learning models using five Arabic corpora specifically annotated for inappropriate speech. By assessing the performance of different models on these corpora, we gain insights into the effectiveness and robustness of the proposed approaches in handling Arabic language-specific challenges.

Moreover, we expand our efforts in this thesis by exploring another area of hate speech detection, where we collect and annotate new data for Algerian hate speech. Furthermore, we evaluate the LSTM models and fine-tuning or Arabic BERT models and compare the performance of these models.

This section presents an overall experimental study to explore and present new sources for Arabic inappropriate speech detection from social media.

3.3.2 Datasets

The objective of our study is to train new DL models to classify Arabic content as whether it contains inappropriate speech. DL models require large annotated data to perform well in the classification tasks, however, hate speech data collection could be challenging regarding the sensitivity of the subject, and annotation complexity. Here in our study, we present To evaluate the approaches, we used four existing datasets that are made available for the research community. As part of this thesis, we collect a new annotated dataset for the Algerian dialect that will be discussed in a separate section.

JCOD We used the dataset presented by (H. Mubarak et al. [2017](#)) which is a collection of an Arabic dataset with the objective of detecting offensive speech. The dataset was derived from comments that were deleted from Aljazira.com, a widely recognized Arabic news platform. To ensure adherence to comment guidelines, which specify the removal of posts containing personal attacks, racism, sexism, or any form of offensiveness, the comments underwent manual moderation. Initially, the authors obtained a substantial amount of data, comprising up to 400,000 comments from approximately 10,000 articles spanning diverse domains. Subsequently, a random selection of 32,000 comments, with lengths ranging from 3 to 200 characters, was chosen to facilitate subsequent annotation. To accomplish this, the selected comments underwent annotation using CrowdFlower, employing three annotators who were tasked with classifying the comments as either obscene, offensive, or clean. Notably, the comments were composed in MSA as well as various dialects.

RCD We used the religious-related hate speech dataset, known as RCD, which was specifically created to address hate speech arising from religious conflicts on Twitter. The RCD dataset, collected by (Albadi et al. [2018](#)), stands as the first Arabic dataset dedicated to religious-based hate content. Notably, this dataset focuses on the four predominant religious beliefs observed in the Middle East region (Islam, Christianity, Judaism, and Atheism). Since Islam is the most practiced religion in this region, the dataset included the two main sects of Islam, namely Sunni and Shia, which comprises 87-90%, and 10-13% of all Muslims, respectively. Here in our study, we only focus if the tweet containing hate speech and neglect the ethnicity and other columns. RCD has 6800 tweets with binary labels.

OSACT-5 We used the *Subtask A* dataset shared within The 4th Workshop on Open-Source Arabic Corpora and Processing tools (OSACT5) (Al-Khalifa et al. 2020). The dataset used in our study contains 7500 tweets that were manually annotated for containing offensive speech or not.

L+T HSAB The L-HSAB (Mulki et al. 2019) and T-HSAB (H. Haddad et al. 2019) are two different datasets that contain Levantine Hate Speech and ABuse and Tunisian Hate speech and ABuse texts, collected from Twitter. L-HSAB dataset combines 5,846 Syrian/Lebanese political tweets, T-HSAB combines 6,024 Tunisian comments, both labeled with Abuse, Hate, and Normal. In this study, we combined the two datasets since they present the same labels to obtain a larger dataset of Tunisian and Levantine Arabic dialects of Abusive and Hate speech.

Table 3.10 shows the number of Inappropriate speech (hate, abusive, offensive) and normal speech in each dataset.

Table 3.10: Hate speech datasets used in our study

| | RCOD | RCD | OSACT | L+THSAB |
|---------------|-------|-------|-------|---------|
| Normal | 5,653 | 3,374 | 5,467 | 11,290 |
| Inappropriate | 5,653 | 2,762 | 1,371 | 6,602 |

Data preprocessing and preparation The writing styles utilized in social media posts often lack uniformity and do not adhere to standard grammar rules, rendering the construction of reliable language models a challenging task. Consequently, we implemented the following normalization procedures on the datasets:

- Links, user mentions, and numbers were normalized.
- Hashtags were removed by deleting the underscore and the # symbol.
- Punctuations, emojis, and words with a length of one were eliminated.
- Word normalization involved reducing the repetition of characters if the repetition count reached three or more.
- Stopwords, which are tokens that do not contribute to the meaning of a sentence, were preserved as they hold significant contextual value in sentiment analysis.

- All non-Arabic scripts were removed from the texts, with the exception of the T-HSAB dataset, which contains Arabizi text.

Recurrent neural networks, like LSTMs, are designed to process sequential data with consistent vector lengths. However, when working with collected texts from social media, it is common to encounter varying text lengths. To address this issue, short texts are padded with special word paddings, such as zeros (0), which do not contribute any meaning.

In our approach, we first identified the longest sentence within each dataset. Then, we padded all the shorter sentences with zero values to ensure that all sentences have the same length as the longest sentence. This padding ensures that all words in the dataset are represented with a uniform vector length, enabling compatibility with LSTM models and facilitating consistent processing of the sequential data.

3.3.3 Proposed models

Attention Mechanism is becoming widely used and is one of the most popular mechanisms in the recent Natural language processing research field. Attention was first introduced for machine translation tasks in (Bahdanau et al. 2014a). The concept of attention in neural networks draws inspiration from the selective focus observed in human visual processing. Our biological systems tend to concentrate on specific elements within a frame or scene while disregarding irrelevant information, thereby aiding perception. In a similar vein, attention mechanisms enable models to dynamically allocate focus to specific portions of input data that are crucial for task performance, while ignoring less relevant parts. This approach enhances processing speed.

The significant progress made in modeling attention within neural networks can be attributed to three main factors:

- Attention is the core mechanism in many SOTA models like BERT (Devlin et al. 2018), Transformer (Vaswani et al. 2017a) and used in different tasks such as image captioning (L. Huang et al. 2019).
- Beside the remarkable performance on the main task, The mechanism also brings an important feature to the world of neural networks which is the interpretability of the results.

- The mechanism was mainly proposed to overcome the limitation of by RNNs when dealing with long input sequences in machine translation.

3.3.4 E-LSOA: An Efficient LSTM-based model

Our proposed model is composed of an **E**mbedding layer, **L**STM layers, and **s**oft **A**ttention layer inspired from (Zhou et al. 2016) which takes the input tokens, learns their word representation, and maps each token to its unique vector representation. The overall model architecture is presented in Fig. 3.10.

Embeddings

Given a sentence consisting of T words $S = \{x_1, x_2, \dots, x_T\}$, every token x_i is converted into a real-valued vector e_i . For each token(tkn) in S , we first look up the embedding matrix $W_{\text{tkn}} \in \mathbb{R}^{d_w \times |V|}$, where V is a fixed-sized vocabulary, and d_w is the size of word embedding. The matrix W_{tkn} is a parameter to be learned, and d_w is a hyper-parameter to be chosen by the user. We transform a word x_i into its word embedding e_i by using the matrix-vector product:

$$e_i = W_{\text{tkn}} v_i \quad (1)$$

where v_i is a vector of size $|V|$ which has value 1 at index e_i and 0 in all other positions. Then, the sentence is fed into the next layer as a set of real-valued vectors $\text{embs} = \{e_1, e_2, \dots, e_T\}$.

LSTM Block

Once we obtain the output vector $\text{embs} = \{e_1, e_2, \dots, e_T\}$ from the embedding layer, it serves as the input sequence for the Long Short-Term Memory (LSTM) process. The LSTM model is a type of recurrent neural network (RNN) that effectively captures sequential information while mitigating the vanishing gradient problem.

To process the input sequence through the LSTM, we iterate over the elements of embs from e_1 to e_T as follows:

Initialize LSTM hidden states: h_0, c_0

For $t = 1$ to T do:

Compute LSTM outputs and hidden states using the equations:

$$f_t = \sigma(W_f \cdot [h_{t-1}, e_t] + b_f)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, e_t] + b_i)$$

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, e_t] + b_c)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, e_t] + b_o)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t$$

$$h_t = o_t \odot \tanh(c_t)$$

In the above equations, $W_f, W_i, W_c, W_o, b_f, b_i, b_c, b_o$ represent the learnable parameters of the LSTM, σ denotes the sigmoid activation function, $[\cdot, \cdot]$ denotes concatenation, and \odot denotes element-wise multiplication. The resulting LSTM outputs h_t and cell states c_t at each time step capture the contextual information from the input sequence. These outputs can be further utilized for downstream tasks such as sentiment analysis, text classification, or any other relevant NLP tasks.

Soft Attention layer

We used the Soft Attention layer proposed for relations classification by (Zhou et al. [2016](#)). Consider the matrix H , composed of output vectors $[h_1, h_2, \dots, h_T]$ generated by the LSTM layer, where T represents the length of the sentence. To form the sentence representation, denoted as r , a weighted sum of these output vectors is computed. This is achieved by applying the hyperbolic tangent function to matrix H , resulting in M . Then, the weights α are obtained by applying the softmax function to the matrix product of a weight vector w and the transpose of M . Finally, the sentence representation r is computed as the matrix product of H and α , transposed. The equations for these computations can be summarized as follows:

$$M = \tanh(H) \quad (3.10)$$

$$\alpha = \text{softmax}(w^T M) \quad (3.11)$$

$$r = H\alpha^T \quad (3.12)$$

Classification In this section, we perform binary classification for all datasets mentioned in section 3.3.2, where we set the binary-cross-entropy as a loss function:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m [t_i \log(y_i) + (1 - t_i) \log(1 - y_i)] \quad (3.13)$$

m denotes the number of training examples. t_{i} represents the true label (target value) for the i -th example. y_{i} denotes the predicted probability (between 0 and 1) of the positive class for the i -th example. The binary cross-entropy loss measures the dissimilarity between the predicted probabilities (y_{i}) and the true labels (t_{i}) for each example. It penalizes the model for larger discrepancies between the predicted and true labels. The loss function aims to minimize this discrepancy during the training process by adjusting the model parameters θ . To study the effectiveness of each component in the proposed ELSoA model (ablative study), we denote in the rest of this section **E-LSTM**, the proposed model without the Soft attention layer, and **ESoA** the proposed model without the LSTM block.

3.3.5 Experiments and results

This section provides a concise summary of the key findings derived from our research endeavor. Initially, we conducted a comparative analysis of the three models across each dataset, considering crucial aspects such as training time and parameter count, which are presented in Table 3.11. Subsequently, we delve into an extensive evaluation of the performance exhibited by each model on diverse datasets. To conduct the experiments, we used a TESLA T4 GPU on Google Colab⁹.

During the training of the models, we employed the Adam optimizer for all the experiments. The models were trained for a total of 50 epochs. The training process was evaluated by

⁹<https://colab.research.google.com/>

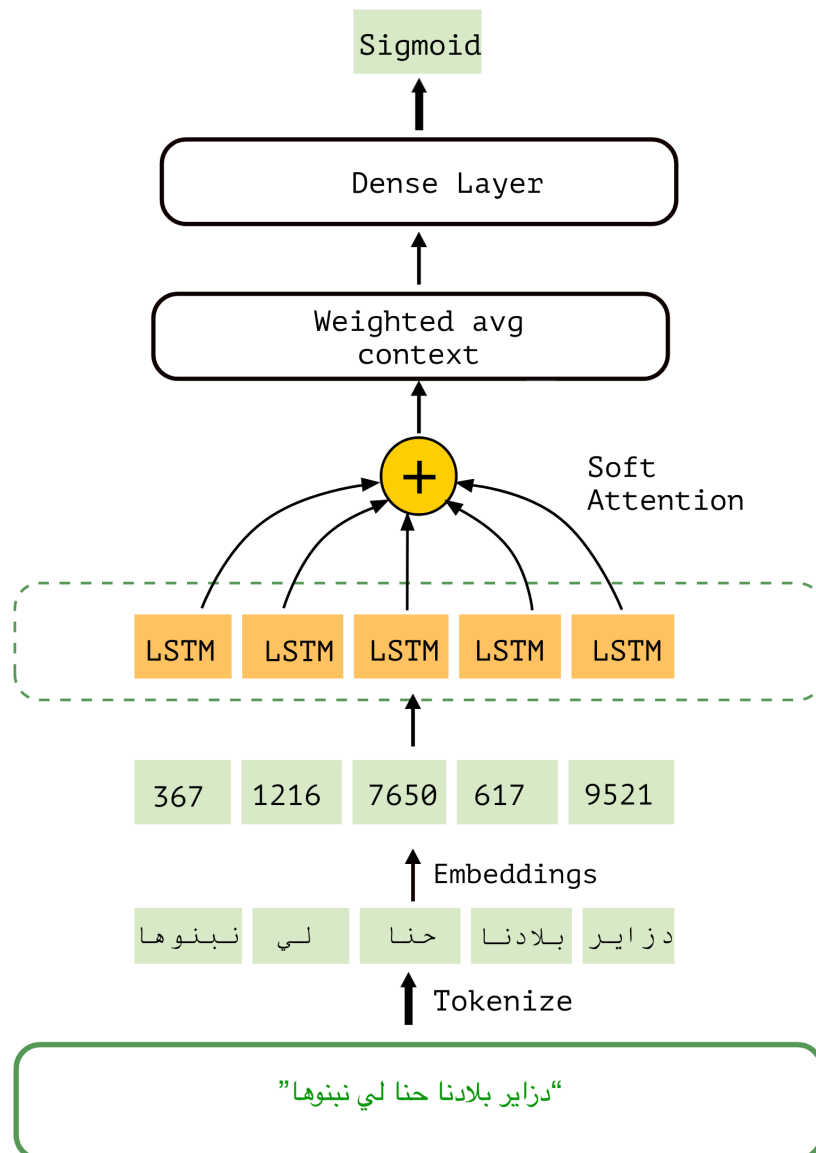


Figure 3.10: Proposed E-LSoA model architecture

measuring the time taken for each epoch in seconds. Additionally, the loss function used for the first three models, which had binary classification tasks, was binary cross-entropy. However, the last model involved three output classes, so we utilized categorical cross-entropy as the loss function.

The performance of the models is reported in three metrics namely Accuracy, Precision, and F1 score.

Experimental settings

In the context of our study, we conducted experiments using various datasets to train and evaluate our models for hate speech detection. Here are the details regarding the datasets and

their configurations:

- RCD: To train the embedding layer, we utilized a subset of 20,000 tokens from the original RCD dataset. The training was performed with a batch size of 62, employing a single neuron in the output layer, and adopting the binary cross-entropy loss function.
- JCOD dataset: Originally comprising three classes, we transformed the dataset for our study by merging the offensive and hate speech classes into a single class. This resulted in a binary classification dataset consistent with the original paper. For training, we used a batch size of 80 and utilized the binary cross-entropy loss function.
- OffensEval 2020 dataset: Due to data availability constraints, the dataset we utilized for training contained 7,500 tweets, which is slightly reduced compared to the 10,000 tweets reported in the competition.
- LHSAB+THSAB dataset: In our study, we combined two datasets presented in section 3, namely LHSAB and THSAB. The objective was to obtain a larger Arabic dialect dataset for hate speech analysis. The resulting combined dataset comprises 11,869 tweets, each labeled as either Abusive, Hate speech or normal text.

By leveraging these datasets with their respective configurations, we aimed to evaluate the performance of our models in hate speech detection and classification tasks.

Table 3.11: Models' comparison across the datasets

| Dataset | Model | # Parameters | Train. time (s) |
|----------|--------|--------------|-----------------|
| JCOD | E-LSTM | 9.8M | 1000 |
| | ESoA | 13M | 150 |
| | ELSoA | 15M | 950 |
| RCD | E-LSTM | 13.1M | 5600 |
| | ESoA | 13M | 150 |
| | ELSoA | 13.8M | 1035 |
| OSACT | E-LSTM | 16.3M | 75 |
| | ESoA | 16M | 50 |
| | ELSoA | 16.6M | 85 |
| L+T HSAB | E-LSTM | 10.8M | 2000 |
| | ESoA | 11.4M | 50 |
| | ELSoA | 11.8M | 2050 |

Table 3.11 presents a comparison of different models across various datasets. The table provides insights into the number of parameters and training times for each model. This information

is crucial for evaluating the performance and efficiency of the models in the context of deep learning.

The datasets included in the comparison are JCOD, RCD, OSACT, and L+T HSAB. For each dataset, three models presented in [3.3.3](#) are compared in terms of size, and training time. The number of parameters listed in the table indicates the complexity of the models. Higher parameter counts generally imply more intricate architectures capable of capturing finer details and nuances in the data. The training time is also provided in seconds, offering insights into the computational resources required for training each model on the respective datasets.

Analyzing the results, it can be observed that across the JCOD dataset, the EL model has 9.8 million parameters and takes 1000 seconds to train. The ESoA model has 13 million parameters and trains in 150 seconds, while the ELSoA model has 15 million parameters with a training time of 950 seconds. Similarly, the comparison is presented for the RCD, OSACT, and L+T HSAB datasets.

This table allows for a comprehensive assessment of the models' performance and resource requirements across different datasets. It serves as a valuable reference for researchers and practitioners in the field of deep learning, aiding in the selection of appropriate models based on their computational efficiency and parameter complexity.

Table 3.12: Experimental results of different models across all datasets

| Dataset | Model | Acc. | Precision | F1-score |
|----------------|--------------|-------------|------------------|-----------------|
| JCOD | E-LSTM | 89.83% | 88.93 % | 89.95% |
| | ESoA | 90.5% | 90.39% | 90.36% |
| | ELSoA | 90.79% | 90.78% | 90.49% |
| RCD | E-LSTM | 96.33 | 97.12 | 97.1% |
| | ESoA | 97.64% | 95.31% | 97.77% |
| | ELSoA | 95.5% | 96.47% | 96.59 |
| OSACT | E-LSTM | 96.33% | 97.12% | 97.1% |
| | ELSoA | 97.47% | 97.77% | 97.17% |
| | ESoA | 97.4% | 97.14% | 97% |
| L+T HSAB | EL | 95.90% | 95.95% | 95.89% |
| | ESoA | 95.67% | 95.77% | 95.67% |
| | ELSoA | 96.18% | 96.14% | 96.09% |

The classification results for hate speech detection using the three models (E-LSTM, ESoA, and ELSoA) are presented in Table [3.12](#). The evaluation metrics used to assess the performance of the models are accuracy, precision, and F1-score.

For the JCOD dataset, the E-LSTM model achieved an accuracy of 89.83%, a precision of 88.93%, and an F1-score of 89.95%. The ESoA model performed slightly better, with an accuracy of 90.5%, a precision of 90.39%, and an F1-score of 90.36%. The ELSoA model showed the highest performance among the three, with an accuracy of 90.79%, a precision of 90.78%, and an F1-score of 90.49%.

Moving to the RCD dataset, the E-LSTM model achieved an accuracy of 96.33%, a precision of 97.12%, and an F1-score of 97.1%. The ESoA model demonstrated higher accuracy at 97.64% but had a slightly lower precision of 95.31%. However, the ESoA model yielded the highest F1-score of 97.77%. The ELSoA model achieved an accuracy of 95.5%, a precision of 96.47%, and an F1-score of 96.59%.

For the OSACT dataset, the E-LSTM model achieved an accuracy of 96.33%, a precision of 97.12%, and an F1-score of 97.1%. The ELSoA model showed the best performance, with an accuracy of 97.47%, a precision of 97.77%, and an F1-score of 97.17%. The ESoA model also performed well, with an accuracy of 97.4%, a precision of 97.14%, and an F1-score of 97%. Lastly, for the L+T HSAB dataset, the E-LSTM model achieved an accuracy of 95.9%, a precision of 95.95%, and an F1-score of 95.89%. The ESoA model demonstrated an accuracy of 95.67%, a precision of 95.77%, and an F1-score of 95.67%. The ELSoA model achieved the highest performance, with an accuracy of 96.18%, a precision of 96.14%, and an F1-score of 96.09%.

The ELSoA (Embedding layer, soft attention, and LSTM) model consistently demonstrates better classification performance compared to the E-LSTM (Embedding layer with LSTM) and ESoA (Embedding layer with soft attention) models across multiple datasets. There are several reasons that can explain the improved performance of the ELSoA model:

Attention Mechanism: The incorporation of the soft attention mechanism in the ELSoA model allows it to dynamically focus on relevant parts of the input sequence, giving more weight to important features. This attention mechanism helps the model to selectively attend to specific aspects that are crucial for hate speech detection, potentially enhancing its ability to capture nuanced patterns and discriminatory language.

Combining Attention with LSTM: The ELSoA model combines the attention mechanism with the LSTM layer. LSTM, a type of recurrent neural network (RNN), is effective in capturing

sequential dependencies and long-term dependencies in the data. By integrating attention with LSTM, the ELSoA model can benefit from both the temporal modeling capabilities of LSTM and the selective focus of attention. This combination allows the model to effectively process and analyze sequential information while focusing on relevant parts.

The improved classification performance of the ELSoA model can be attributed to several factors. Firstly, the model exhibits increased complexity compared to the E-LSTM and ESoA models, as indicated by its higher number of parameters. This augmented complexity enables the ELSoA model to capture intricate patterns and representations within the data, potentially enhancing its discriminatory power for hate speech detection.

Additionally, the ELSoA model leverages an embedding layer, LSTM layer, and soft attention layer, facilitating improved representation learning. The embedding layer transforms input tokens into continuous vector representations, effectively capturing semantic and contextual information. Complementing this, the LSTM layer captures sequential dependencies and contextual information across the input sequence. By incorporating both layers, the ELSoA model can acquire more informative and discriminative representations, leading to enhanced classification performance.

Furthermore, the ELSoA model may have benefitted from dataset-specific adaptation. It is important to acknowledge that different datasets possess unique characteristics and linguistic nuances. The ELSoA model might have been particularly well-suited for capturing the specific patterns and characteristics present in the datasets used for evaluation. This alignment between the model and the dataset attributes likely contributed to the improved classification performance observed on those specific datasets.

The superior performance of the ELSoA model in hate speech detection can be attributed to its increased model complexity, which allows for the capture of intricate patterns and representations. Moreover, the combination of an embedding layer and an LSTM layer enables improved representation learning, leading to more informative and discriminative representations. Additionally, the model's potential adaptation to the specific characteristics of the datasets used for evaluation further contributes to its enhanced classification performance.

3.3.6 Hate speech detection in Algerian dialect

In this section, we present a new dataset for offensive and hate speech detection in the Algerian dialect. The dataset, named *OHAD* (Offensive and Hate Speech in Algerian Dialect), consists of 13,000 sentences collected from Algerian communities and pages on social media platforms such as Facebook and Twitter. The dataset is aimed at addressing the critical issue of hate speech and offensive content on social media platforms in the Algerian context.

To ensure the quality and accuracy of the annotation, the OHAD dataset was annotated by two native speakers of Algerian Arabic. The annotators were given clear instructions and guidelines on the annotation process to ensure consistency in their annotations. The annotators were also trained to identify the presence of any kind of inappropriate speech in the sentences including; offensive and hate speech, such as insults, threats, racism, and discrimination, among others.

The annotation process of OHAD involved reading and analyzing each sentence in the dataset, identifying whether it contained any form of offensive or hate speech, and labeling it accordingly. The annotators were required to label each sentence as either "offensive" or "normal speech." To ensure the quality of the annotations, the annotators cross-checked each other's work and resolved any discrepancies through discussion. We ensured the balance between the two labels of the dataset

After the annotation process was completed, the OHAD dataset was split into training, validation, and testing sets. The training set was used to train offensive and hate speech detection models, while the validation set was used to fine-tune the models and select the best-performing one. The testing set was used to evaluate the performance of the selected model.

OHAD dataset is a new resource for offensive and hate speech detection in the Algerian dialect. The dataset was carefully annotated by two native speakers of Algerian Arabic, and the annotation process was designed to ensure consistency and accuracy. The availability of such a dataset will enable researchers and practitioners to develop effective models for detecting offensive and hate speech in Algerian social media communities, and ultimately help to promote a safer and more respectful online environment.

Experiments and results

We put the collected OHAD dataset into an experimental evaluation of multiple deep learning models using different contextual representations, namely neural embedding models and BERT-

based models.

The NN models are: BiGRU, BiLSTM and 1D-CNN, using learnable embeddings from the TensorFlow library. Also, We put multiple BERT-based models namely AraBERT (Antoun et al. 2020), MARBERT (Abdul-Mageed et al. 2021), and DziriBERT (Abdaoui, Berrimi, et al. 2021). The results are reported in terms of binary accuracy.

Table 3.13: Performance Comparison of Various Models for Algerian hate speech detection

| Model | Acc | Prec | F1 |
|-----------|--------------|--------------|--------------|
| BiLSTM | 86.57 | 86.61 | 86.32 |
| BiGRU | 86.74 | 86.35 | 86.64 |
| CNN | 86.12 | 86.46 | 86.14 |
| AraBERT | 92.76 | 92.83 | 92.96 |
| DziriBERT | 96.62 | 96.81 | 96.97 |
| MARBERT | 95.87 | 95.81 | 95.94 |

Discussion

The table provided in Table 3.13 presents the classification performance of various neural network models and BERT-based models for the identification of offensive speech in the Algerian dialect. Among the neural network models, BiGRU demonstrates the highest accuracy at 86.74% and F1 score at 86.64%, closely followed by BiLSTM with an accuracy of 86.57% and F1 score of 86.32%. The CNN model exhibits slightly lower performance with an accuracy of 86.12% and an F1 score of 86.14%.

When examining the BERT-based models, DziriBERT, which is specifically pretrained on the Algerian dialect, outperforms the other models with the highest accuracy (96.62%), precision (96.81%), and F1 score (96.97%). This suggests that DziriBERT is well-suited to handle the nuances and peculiarities of the Algerian dialect, resulting in superior performance for offensive speech identification.

MARBERT and AraBERT also exhibit strong performance, with MARBERT achieving an accuracy of 95.87%, precision of 95.81%, and an F1 score of 95.94%. MARBERT is trained on a larger dataset compared to AraBERT, which likely contributes to its higher performance. AraBERT, on the other hand, yields an accuracy of 92.76%, precision of 92.83%, and an F1 score of 92.96%.

In conclusion, the BERT-based models, particularly DziriBERT, show considerable promise for

the task of identifying offensive speech in the Algerian dialect. The specialized pretraining of DziriBERT on the Algerian dialect and the larger dataset used for training MARBERT both contribute to their superior performance compared to the other models.

3.3.7 Conclusion

This section delves into a distinct research domain of Arabic text analysis, focusing on the detection and classification of inappropriate, abusive, and hate speech prevalent in Arabic social media platforms. In the initial part of this section, we conduct an extensive investigation of diverse forms of inappropriate content across Tunisian, Levantine, MSA, and Khalidji dialects. We employ a novel LSTM model integrated with a soft attention mechanism, which exhibits remarkable classification performance validated on four well-established datasets.

Moving forward, the subsequent part of this section introduces a novel annotated corpus specifically curated for Algerian hate speech, collected from social media platforms like Twitter and Facebook. Within this subsection, we validate advanced techniques and employ effective word representation methodologies. Notably, we employ DziriBERT, a BERT-like model pretrained on Algerian text, which emerges as the top performer in terms of hate speech detection accuracy among various classifiers.

The proposed methodologies presented in this section not only yield novel results but also pave the way for future research endeavors in this burgeoning field. These findings open up new avenues for exploration and advancement in the domain of Arabic text analysis, particularly in addressing the challenges associated with inappropriate, abusive, and hate speech.

3.4 Chapter conclusion

This chapter presents three main contributions towards analyzing Arabic text and speech using deep learning models. Firstly, a new attention mechanism-based deep learning approach is proposed and tested for Arabic sentiment analysis using publicly available datasets. The model assumes two types of embeddings - FastText and Learnable embeddings - and employs Bidirectional GRU/LSTM to capture contextual information. The study also showcases the performance of various embedding strategies, and a comparison between baseline models revealed the simplicity of GRU cell concept compared to LSTM. The proposed model outperformed both baseline models and state-of-the-art models reported in the original references of the datasets.

Secondly, the chapter describes a novel study that aims to build and analyze an emotional speech corpus of the Algerian dialect and to propose a new hybrid classification model to recognize emotions from Arabic speech. A large annotated dataset consisting of 1202 audio records annotated as happy, angry, neutral, or sad was collected and constructed. Various experiments were conducted using machine learning classification algorithms and deep convolutional and recurrent neural networks. The proposed LSTM-CNN model outperformed all other classifiers and approaches with a high accuracy of 93.34%. The results showed that the use of LSTM networks can give interesting results in speech emotion recognition, especially in our case study of emotion detection from the Algerian dialect.

Lastly, the chapter highlights the critical issue of hate speech and offensive content on Arabic social media communities and proposes a novel deep learning architecture based on the attention mechanism for smooth and accurate learning and classification. The proposed attention-based models outperformed other architectures significantly in terms of performance and processing time across four datasets.

Chapter 4

Arabic dialect identification

In this chapter, we delve into the complexities of the Arabic language, specifically addressing the challenges posed by the wide range of dialects spoken across different regions worldwide, and how to leverage ML models to identify these dialects. Recognizing and distinguishing between various dialects is critical, as they can significantly impact the interpretation of textual content. Such dialects manifest as unique combinations of pronunciation, vocabulary, grammar, and syntax, often leading to disparate words, expressions, and meanings for identical concepts. This, in turn, can result in misinterpretations or misunderstandings of the context. Our research contributions, as detailed in this chapter, revolve around the identification of multiple Arabic dialects using advanced NLP techniques, effective word representations, and efficient deep learning models.

4.1 Introduction

One of the key challenges in NLP is the processing and understanding of language variations, including dialects. Arabic, as a widely spoken language, is characterized by a rich diversity of dialects, which adds complexity to the development and evaluation of NLP systems tailored to the Arabic language.

Dialect identification (DI) plays a vital role in enhancing the performance of various NLP tasks such as machine translation (Harrat et al. [2019](#)), sentiment analysis (Boudad et al. [2018](#)), and named entity recognition (A. H. Dahou et al. [2023b](#)). By accurately identifying the dialect of a given text, NLP systems can employ dialect-specific models or techniques, leading to improved performance and a more nuanced understanding of the linguistic context. Furthermore,


the identification of Arabic dialects is particularly important for social media analysis, where dialectal variations are often prevalent and may significantly influence the interpretation of user-generated content.

To this end, the development of robust and efficient dialect identification systems for Arabic is an essential step toward unlocking the full potential of NLP applications in the Arab world. By addressing the challenges posed by dialectal variations, researchers and practitioners can enable more accurate and context-aware NLP systems, ultimately contributing to a deeper understanding of the Arabic language and its diverse expressions.

In this chapter, we make two primary contributions to the field of Arabic dialect identification: Firstly, we introduce a novel corpus specifically tailored for the identification of North-African dialects, also known as Maghrebi, and secondly, we explore various learning approaches for the classification task, including classical machine learning methods, neural network-based models, and fine-tuning of pre-trained language models. Our goal is to deepen our understanding of the intricacies of Arabic dialects and develop more accurate techniques for their identification.

We focus on the Maghrebi dialect, as it is widely used on social media by users from countries such as Algeria, Morocco, and Tunisia. These individuals often prefer to communicate in their dialect rather than in MSA. Moreover, Arabizi, particularly French-Arabizi, is a common form of communication in these regions.

The Maghreb region exhibits a diverse range of dialects, making their differentiation a challenging task. For instance, Algeria, located at the center of the Maghreb region, showcases dialectal variations across its eastern and western regions. The Tunisian dialect is predominantly spoken in the east, while the Moroccan dialect prevails in the west. These variations in dialects further complicate the process of dialect identification in the Maghreb region.

This graphic illustrates the various dialects spoken in Maghreb countries, where individuals residing in border regions tend to communicate in a similar dialect to their neighboring country. The image showcases different expressions for the question "what are you doing" in various Maghrebi dialects. 

¹<https://www.qfi.org/blog/infographic-dialects-arab-world/>

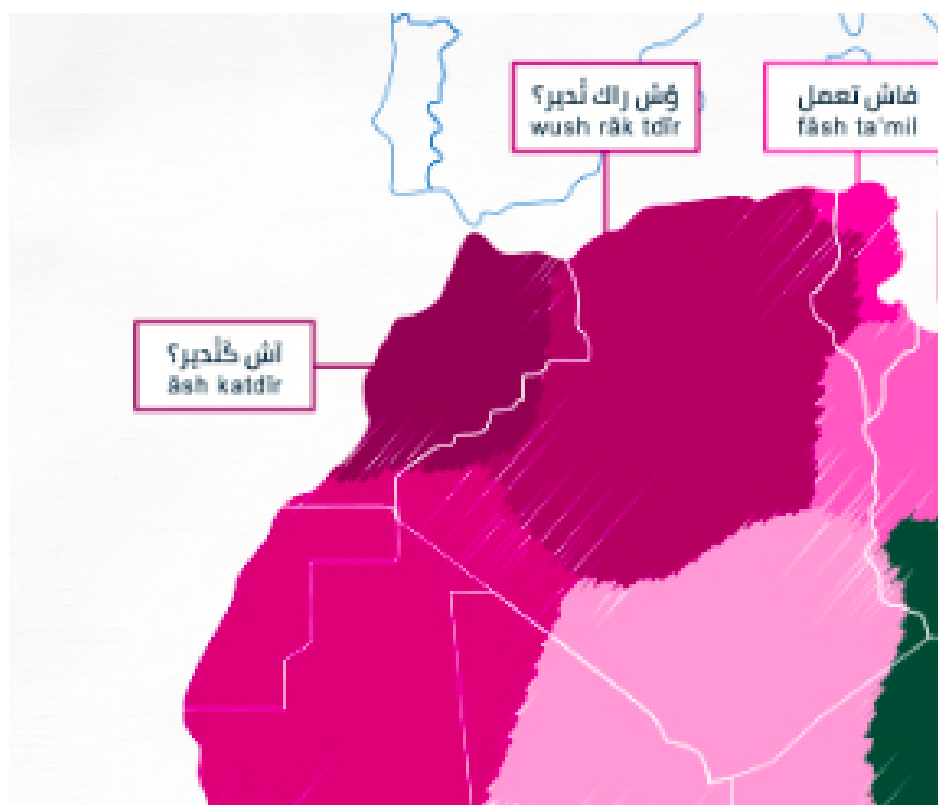


Figure 4.1: Maghrebi dialects variations.

4.2 Methodology

The main objective of this chapter is to explore the effectiveness of diverse classifiers in identifying Maghrebi dialects. Our methodology entails an extensive evaluation of various classifier types, encompassing not only traditional machine learning methods like logistic regression and support vector machines (SVM) that utilize handcrafted TF-IDF features but also neural network-based models such as LSTMs and CNNs. In addition to these techniques, we investigate the performance of Arabic BERT-based models.

To ensure a thorough analysis, we design a series of experiments that compare the accuracy, precision, and F1-score metrics of the different classifiers. Moreover, we analyze the effect of hyperparameter tuning on the performance of each model and discuss the implications of these findings on the overall performance. By doing so, we aim to not only identify the most effective classifier for Maghrebi dialect identification but also to gain a deeper understanding of the underlying linguistic features that contribute to accurate dialect classification.

Furthermore, we explore the potential of transfer learning by fine-tuning pre-trained Arabic BERT-based models on our novel dataset. This approach allows us to assess the adaptability

of these models to the specific task of Maghrebi dialect identification and to determine their effectiveness in comparison to other classification techniques. Through this comprehensive analysis, we seek to provide valuable insights and contribute to ongoing efforts in the field of Arabic dialect identification in NLP systems.

4.2.1 Dataset

Social media platforms use have witnessed a surge in popularity in North Africa, amassing a considerable user base in Algeria, Tunisia, and Morocco. To examine the prevalence of local dialects on this platform, we carried out an investigation to identify the most prominent Facebook pages within these communities, characterized by follower counts surpassing 100,000. Our analysis concentrated on posts and comments composed in regional dialects.

Subsequent to categorizing Facebook pages by country, we employed a manual data collection approach to collect approximately 20,000 posts and comments from these pages, while ensuring that the user names were excluded from the scrapping process. The collected Maghreb-DI dataset was then labeled in accordance with its corresponding group, culminating in a total of 60,000 sentences evenly distributed across three balanced classes. This comprehensive dataset enabled us to rigorously assess the performance of various classifiers and techniques in accurately identifying North-African dialects within the context of social media communications.

Table 4.2 and 4.3 show samples in Maghrebi dialects, where sentences are written in Arabizi (Arabic with Latin characters) and in Arabic.

Table 4.1: Number of sentences per dialect in Maghreb-DI dataset

| | Algerian | Moroccan | Tunisian |
|-------------------|-----------------|-----------------|-----------------|
| Num. of sentences | 21230 | 20150 | 19050 |

Social media data is often characterized by the presence of various forms of noise, including special characters, repeated words, URLs, emoticons, punctuation marks, and extraneous words. To improve the quality of the collected text sentences and minimize the noise, we implemented a series of cleaning functions on the data. Specifically, we removed numerical characters, URLs, and hashtags by eliminating the ”#” symbol. Additionally, we discarded special characters, such as punctuation marks and Arabic diacritics. This comprehensive preprocessing approach ensures a cleaner and more consistent dataset, which is crucial for the subsequent analysis and evaluation of dialect identification methods.

Table 4.2: Samples of sentences written in Arabizi.

| Language | Arabizi | English (<i>translation</i>) |
|----------|----------------------------------|--|
| Algerian | wech kho rak tconcté b PC te3ek? | dude, are you connecting on your laptop? |
| Tunisian | yezzi berka m tfadlek 3liya | stop playing with me |
| Moroccan | wakha a sahbi, ka nkelmek mb3d | ok my friend, I will call you later |

Table 4.3: Samples of sentences written in Maghrebi dialect.

| Language | Arabizi | English |
|----------|---|--|
| Algerian | معلاباليش هذا العبياد كيفاش يخمو يا شكوي تقول معندهمش رأس | I don't know how these people think, they are insane. |
| Tunisian | المهم قالوها كهو باش نمشو للبلاد | The important thing is that they said it, so we can go home. |
| Moroccan | باباك راه ناعس و مريض منبغيش نصدعوه صافي | Your dad is sleeping and sick, we don't want to bother him. |

4.2.2 Data Preparation

Before training the neural network models, we prepared the dataset by performing several preprocessing steps, including tokenization, padding, and splitting the data into training and validation sets. To ensure uniform input length for the neural networks, we applied padding to the sequences, effectively truncating or zero-padding them to a fixed length.

We divided the dataset into a training set and a validation set, typically following an 80%-20% split. The training set was used to train the models, while the validation set was employed to evaluate the performance of the models during the training process and fine-tune the hyperparameters.

4.3 ML-based model for Arabic DI

In this study, we employ classical ML algorithms, including SVM, Logistic Regression (LR), and Multinomial Naive Bayes (NB) models, for DI, as presented in Fig. 4.2. These ML-based models are chosen due to their proven track record in various natural language processing tasks, including text classification and sentiment analysis.

To represent the text data and extract meaningful features for our ML-based models, we utilize the Term Frequency-Inverse Document Frequency (TF-IDF) technique. TF-IDF is an established method for evaluating the importance of a term within a document relative to a collection of documents. It assigns a numerical weight to each word in a document, which is proportional to the term's frequency in the document and inversely proportional to the number of documents containing the term. This approach addresses the limitations of the Bag of Words model, which may give undue importance to frequently occurring words that are less informative.

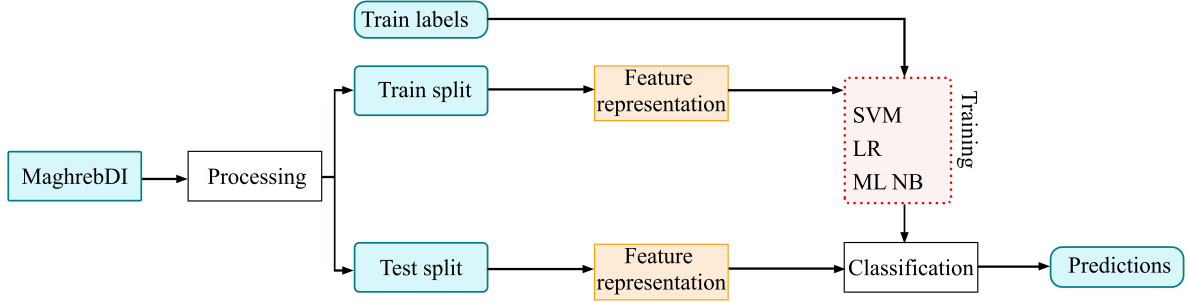


Figure 4.2: Generic graph showing the overall DI methodology for classical ML experiments

The TF-IDF technique effectively captures the significance of words within the document, giving higher weights to terms that are more relevant to the document and lower weights to more common words. The formula for calculating the TF-IDF value for term t in document d is as follows:

$$Tf-idf_{t,d} = (1 + \log tf_{t,d}) \cdot \log \frac{N}{df_t}$$

where:

tf : number of occurrences of t in d (4.1)

df : number of documents containing t

N : total number of documents

By employing the TF-IDF technique, we can effectively represent the text data and extract valuable features for our ML-based models, enabling them to make accurate predictions in the task of dialect identification.

Table 4.4: SVM and Logistic Regression model hyperparameters

| Model | Kernel | Class weight | Penalty | C (reg. term) |
|---------------------|--------|--------------|---------|---------------|
| SVM | Linear | - | l2 | 1 |
| Logistic Regression | - | Balanced | l2 | 10 |

In the present experiments, the Multinomial Naive Bayes classifier was utilized with its default parameters, as presented in Table [4.5](#).

Table 4.5: Default parameters and values for Multinomial Naive Bayes classifier in scikit-learn

| Parameter | Default Value |
|-------------|---------------|
| alpha | 1.0 |
| fit_prior | True |
| class_prior | None |

4.4 Neural network-based models Arabic for DI

In this section, we present the methodology for the neural network-based experiments conducted to evaluate the performance of deep learning models on the task of dialect identification. Our methodology encompasses two primary neural network architectures: Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNNs).

4.4.1 Model Architectures

LSTM

The LSTM model is a type of RNN designed to handle long-term dependencies in sequential data. In our experiments, we constructed an LSTM-based architecture, as illustrated in Fig. 4.3, which included an embedding layer, one or more LSTM layers, and a dense output layer with a softmax activation function for dialect classification. We experimented with various hyperparameters, such as the number of LSTM units, dropout rates, and learning rates, to optimize the model's performance.

1D-CNN model

The CNN model is a popular neural network architecture for processing grid-like data, such as images and text. In our experiments, we designed a CNN-based architecture specifically for text classification. The architecture comprised an embedding layer, one or more convolutional layers followed by pooling layers, and a dense output layer with a softmax activation function for dialect classification. We explored various hyperparameters, including the number of filters, filter sizes, and pooling strategies, to optimize the model's performance.

4.4.2 Training and Evaluation

We trained the LSTM and CNN models using the prepared training set, employing an appropriate loss function, such as categorical cross-entropy, and Adam optimization algorithm. We

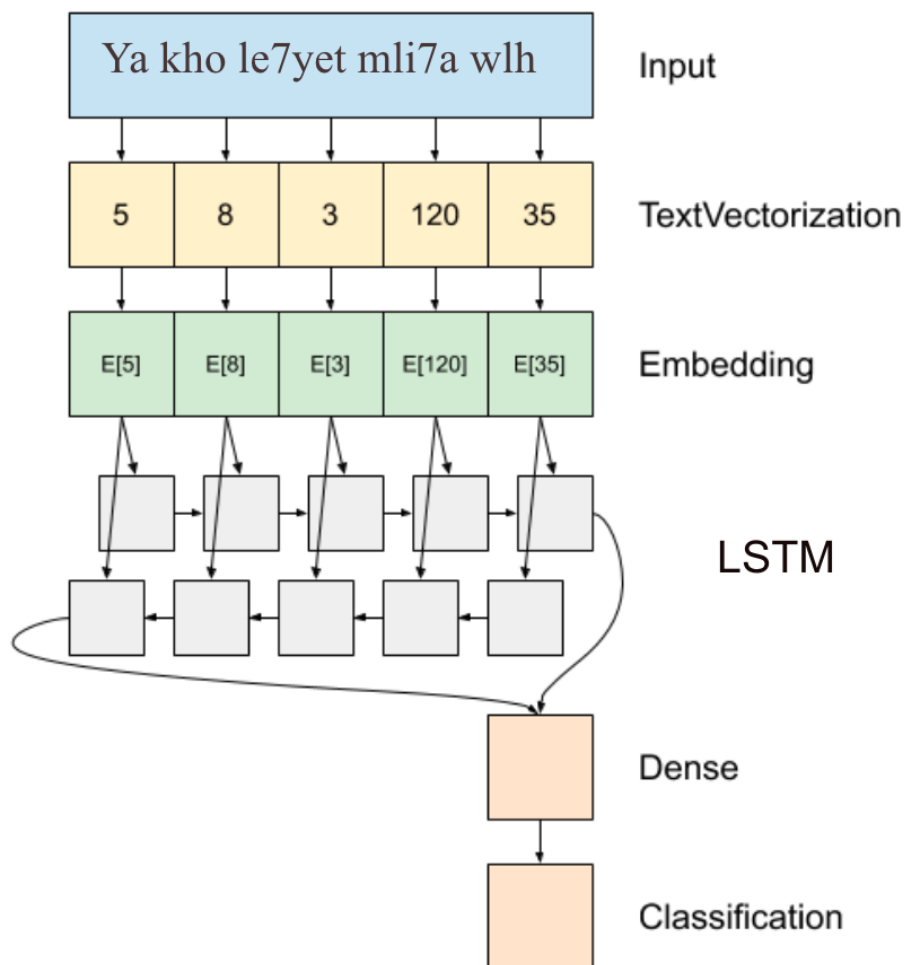


Figure 4.3: Generic graph for Maghreb DI using an LSTM model

monitored the performance of the models on the validation set during training and adjusted the hyperparameters as necessary to prevent overfitting and achieve optimal performance.

After training, we evaluated the final performance of the LSTM and CNN models on a separate test set, comparing their performance to the traditional machine learning approaches and Arabic-specific models to determine the most effective technique for Maghrebi dialect identification.

4.5 Bert-based models for Arabic DI

Expanding upon the work outlined in our conference publication (Berrimi, Abdelouaheb Mousaoui, Oussalah, et al. [2020a](#)), this section delves deeper into the potential of state-of-the-art Arabic pre-trained language models (LLMs) for the task of Arabic dialect identification (AD). In an effort to thoroughly assess their performance, we present a comprehensive list of pre-trained

language models and evaluate their effectiveness across two distinct benchmarks. We analyze and discuss a variety of related papers to provide a broader context for our findings and ensure a comprehensive understanding of the current state of the field.

In this study, we fine-tune multiple Arabic BERT-based models to assess the capability of these pre-trained language models for the task of identifying Arabic dialects. These models, which include AraBERT (Antoun et al. 2020), DziriBERT (Abdaoui, Berrimi, et al. 2021) and MARBERT (Abdul-Mageed et al. 2021), have been pre-trained on extensive Arabic text corpora that encompass dialectal texts from the countries of interest, such as Algeria, Tunisia, and Morocco. By fine-tuning these models on our labeled dataset, we can evaluate their performance in accurately identifying the dialects present in the North-African context.

Furthermore, we explore the influence of model architecture, training data size, and the pre-training process on the performance of Arabic BERT-based models. We also investigate the impact of various hyperparameter settings on the fine-tuning process and the overall effectiveness of these models for Arabic dialect identification. Through this comprehensive analysis, we aim to identify the optimal combination of model architecture, pre-training strategies, and fine-tuning parameters that yield the best performance for the task at hand.

By examining the performance of Arabic BERT-based models in identifying dialects, we hope to contribute valuable insights to the ongoing research efforts in Arabic dialect identification, ultimately aiding in the development of more effective and efficient techniques for addressing the unique challenges posed by Arabic dialects within the realm of natural language processing.

4.6 Experiments and results

In the experimental section of this chapter, we detail the process of evaluating the effectiveness of various classifiers and models for Arabic dialect identification. We first describe the preprocessing steps applied to the dataset, which include tokenization, stopword removal (this was only applied for classical ML algorithms since TF-IDF features are usually impacted by stopwords), and feature extraction. Next, we outline the parameters and configurations of the different classifiers, including traditional machine learning algorithms, neural network-based models, and Arabic BERT-based models. The classification results for dialect identification using different models are summarized in Table 4.6. As the number of features increases, we

Table 4.6: Experimental results of ML models based on the number of features

| Number of Features | SVM | NB | LR |
|--------------------|--------|--------|--------|
| 5,000 | 77.21% | 70.42% | 77.14% |
| 1,0000 | 85.43% | 80.11% | 85.75% |
| 20,000 | 90.3% | 83.54% | 87.79% |
| 30,000 | 92.88% | 86.32% | 89.19% |
| 40,000 | 93.55% | 89.41% | 90.75% |

observe a consistent improvement in classification performance across all three classifiers. For SVM, the accuracy starts at 77% with 5000 features and gradually improves to 93.55% with 40000 features. This demonstrates the effectiveness of SVM in capturing complex patterns and discriminating between different dialects as more informative features are included. NB classifier shows a similar trend, with an accuracy of 70% for 5000 features and reaching 89.41% with 40000 features. The NB classifier, although relatively simple, performs reasonably well in dialect identification, especially when provided with a larger number of features. LR classifier achieves accuracies of 77% to 90.75% across the range of feature counts. LR shows consistent improvement as the number of features increases, highlighting its capability to learn from a larger set of features and make better dialect predictions. Overall, increasing the number of features leads to improved accuracy for all three classifiers. The SVM classifier consistently achieves the highest accuracy among the three models, while NB and LR classifiers also exhibit competitive performance. These results suggest that the ML models, particularly SVM, NB, and LR, are effective in dialect identification and can benefit from the inclusion of a larger number of features.

We then present our experimental results by comparing the performance metrics of each classifier, such as accuracy, F1-score, and precision, on the task of North-African dialect identification. Additionally, we analyze the impact of hyperparameters and model architecture choices on the overall performance of these models. In our discussion, we provide insights into the strengths and weaknesses of each approach, highlighting the most effective techniques for Arabic dialect identification based on our findings.

4.7 Discussion

The results presented in Table [4.7](#) demonstrate the varying performance levels of different natural language processing models applied to the task of dialect identification. A comprehensive

Table 4.7: Performance comparison of different models using Accuracy, F1-score, and Precision on Maghreb-DI

| Group | Model | Accuracy | F1-score | Precision |
|-------------------|-----------|--------------|--------------|--------------|
| ML classifiers | NB | 89.41 | 90.13 | 91.12 |
| | LR | 90.75 | 90.90 | 91.36 |
| | SVM | 93.55 | 92.95 | 92.45 |
| NN models | LSTM | 94.51 | 95.61 | 95.91 |
| | CNN | 96.21 | 96.64 | 95.95 |
| BERT-based models | AraBERT | 96.53 | 96.95 | 96.86 |
| | MARBERT | 97.83 | 97.19 | 97.98 |
| | DziriBERT | 97.16 | 97.39 | 97.83 |

analysis of these results reveals the following insights:

Classical ML algorithms, which include Naive Bayes (NB), Logistic Regression (LR), and Support Vector Machines (SVM), exhibit a moderate performance with accuracy scores ranging between 90 and 93.55. These models employ traditional machine learning techniques and rely on handcrafted features, such as TF-IDF, to achieve satisfactory results. However, their performance is limited due to the inherent challenges of handling complex language patterns and semantics in dialect identification tasks.

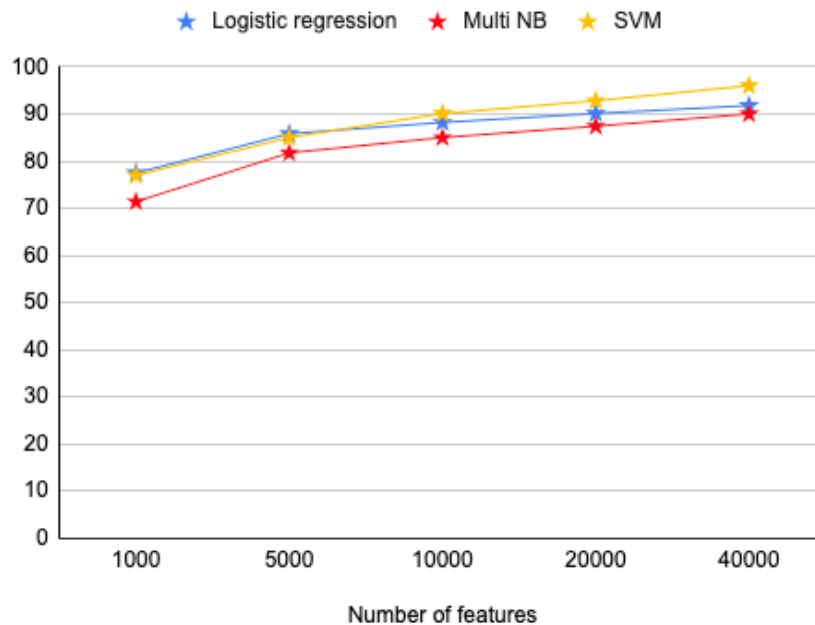


Figure 4.4: Accuracy level according to the number of features used in TF-IDF without stop-words.

Neural network-based models, specifically LSTM and CNN, demonstrate improved performance compared to classical ML algorithms, with accuracy scores ranging between 93 and 95. These

models leverage deep learning techniques to automatically learn intricate language patterns and representations, resulting in more robust performance for dialect identification. Nevertheless, they may still struggle with certain linguistic nuances and contextual understanding, which can limit their overall performance.

BERT-based models, including AraBERT, MARBERT, DziriBERT, outperform both classical ML algorithms and neural network-based models, achieving accuracy scores above 96. These models benefit from the powerful pre-training and fine-tuning capabilities of the BERT architecture, which allows them to learn rich language representations and contextual understanding. Also, MARBERT achieved the highest performance score, this was due the large and variety of dialects in its pretraining corpus. As a result, BERT-based models excel at capturing the subtleties and complexities of dialects, leading to superior performance in dialect identification tasks.

multiple Arabic BERT-based models, including AraBERT, DziriBERT, and MARBERT have been put through experimental comparison.

4.8 Conclusion and future work

This chapter addresses the challenging task of Arabic dialect identification and proposes multiple classification models to improve its efficiency. To this end, we present a novel database consisting of North African text data and conduct a comparative study of classical ML models using TF-IDF features with various parameters, neural-networks bade models, and BERT-based models. Our results demonstrate that despite the close similarity between the text samples of different dialects, the ML models show promising performance in identifying them.

To expand our research, we further investigate and compare additional benchmarks and robust classifiers to identify more diverse Arabic dialects, including Egyptian, Gulf, and Saudi dialects, among others. Our analysis reveals that Arabic BERT models exhibit superior performance in identifying diverse dialects, primarily due to their robust word representation capabilities.

Indeed, the word embeddings generated by Arabic BERT models are highly effective in capturing the nuanced meaning and context of the dialectal words and phrases. This enhances the overall accuracy of the classification models and results in better performance in identifying diverse Arabic dialects. Our research contributes to advancing the field of Arabic dialect identification

by introducing novel datasets, exploring various classification models, and providing valuable insights into the effectiveness of different approaches.

In our future work, we plan to extend our research efforts by expanding the scope of our dataset. Specifically, we aim to collect a more comprehensive dataset that includes region-based data for each country, capturing the rich diversity of dialects within each region. By doing so, we can ensure a more accurate representation of the linguistic variations present in Arab countries.

Additionally, we are eager to delve deeper into the identification of Arabic dialects from speech records by employing advanced deep learning models. These models have shown great potential in various natural language processing tasks, and we believe they can provide valuable insights and improved accuracy in dialect identification.

By pursuing these avenues of research, we aim to enhance our understanding of Arabic dialect identification and contribute to the development of more effective and robust techniques in this field.

Chapter 5

DziriBERT: A Language model for Algerian dialect

Pre-trained transformers are now the de facto models in NLP given their state-of-the-art results in many tasks over various languages. However, most of the current models have been trained on languages for which large text resources are already available (such as English, French, Arabic, etc.). Therefore, there are still a number of low-resource languages that need more attention from the community. In this chapter, we study the Algerian dialect which has several specificities that make the use of Arabic or multilingual models inappropriate. To address this issue, we collected more than one million Algerian tweets and pre-trained the first Algerian language model: DziriBERT. When compared with existing models, DziriBERT achieves better results, especially when dealing with the Roman script. The obtained results show that pre-training a dedicated model on a small dataset (450 MB) can outperform existing models that have been trained on much more data (hundreds of GB). Finally, our model is publicly available to the community [\[1\]](https://huggingface.co/alger-ia/dziribert).

5.1 Introduction

Neural word embeddings such as FastText (Tomas Mikolov, Grave, et al. [2018](#)), Word2vec (Tomas Mikolov, Sutskever, et al. [2013](#)), and Glove (Pennington et al. [2014](#)) have proven to be effective in encoding tokens from the corpus into meaningful vectors, which can be used as

¹<https://huggingface.co/alger-ia/dziribert>

the first layer before training task-specific models. However, these models suffer from major drawbacks, including the "out of vocabulary" problem resulting from generating and mapping each word to a specific vector, as well as the inability to incorporate contextual information in which a word appears into its embedding.

In recent times, there has been a growing interest in pre-training and fine-tuning large language models based on the transformer architecture (Vaswani et al. 2017a; Devlin et al. 2019; Brown et al. 2020). These models have shown significant improvements in various NLP tasks such as Text Generation (Brown et al. 2020), Sentiment Analysis (Berrimi, Oussalah, et al. 2022), Text Classification (Berrimi and Abdelouaheb Moussaoui 2020), Named Entity Recognition (Brandsen et al. 2022), and Text Summarization. Unlike earlier word embeddings like Word2vec and Glove (Tomas Mikolov, Sutskever, et al. 2013; Pennington et al. 2014), contemporary language models are designed to generate contextualized embeddings, which has resulted in a major quality leap in most Natural Language Processing tasks. The vector representation of a word is now solely based on the context in which the word is used, which includes both left and right contexts.

However, most of the current transformers have been pre-trained on languages for which large text resources are already available, such as English (Devlin et al. 2019), French (Martin et al. 2020) and Italian (Polignano et al. 2019). Even multilingual models, such as the mBERT (Devlin et al. 2019) and XLM-R (Conneau et al. 2020), are limited to official languages that have a large web presence. Low-resource languages such as African and Arabic dialects received less attention due to the lack of data and their specific and/or complex morphology. For example, the Algerian dialect is spoken by 44 Million people but lacks publicly available datasets. In addition to the challenges imposed by low resource languages, pretraining Transformers models come with several challenges, namely the huge amount of training data is required, and as a consequence hundreds of GPUs.

Indeed, MSA is the most common written language in official documents, books, and newspapers in Algeria. However, the local dialect is very frequent in informal communications, messaging, or in the social media sphere. A recent study (Younes et al. 2020) showed that 74.6% of the Algerian web-generated content (mostly on Facebook) is conveyed in dialectal Arabic rather than MSA, and 62% of this content is transcribed in Roman alphabet characters (which is also known as Arabizi).

The Algerian dialect is mainly inspired from standard Arabic but also from Tamazight², French, Turkish, Spanish, Italian, and English. It has several specificities that make the application of MSA or multilingual models inappropriate. First, it may be written either using Arabic or Roman letters (e.g. Salam سلام (eng: Peace)). Then, numbers are sometimes used to represent letters that do not exist in the Roman alphabet (e.g. the use of the number 3 to represent the letter ع or the number 7 to represent the letter ح). Finally, despite the influence of the above-cited languages, the Algerian dialect also has its own vocabulary that does not exist in other standard languages.

In this chapter, we present a new BERT-like model for the Algerian dialect, named DziriBERT. It has been pre-trained on one Million Algerian tweets. We evaluate DziriBERT on sentiment, emotion and topic classification datasets. The experiments revealed that DziriBERT achieves new state-of-the-art results on several datasets when compared to existing Arabic and multilingual models.

Our contributions can be summarized as follows:

- We present a new developed BERT model for the Algerian dialect that was pretrained on a new scraped data.
- We present Dziri-Corpus: a newly scraped Algerian dialect dataset that contains 3.2M text data scraped from multiple social media websites.
- We evaluate DziriBERT on two downstream tasks: sentiment Analysis on two datasets and offensive language detection. The DziriBERT model presents new SOTA results compared to mBERT , xlm-RoBERTa, Arabert and MARBERT.
- We release DziriBERT on popular deep learning frameworks.

The rest of the chapter is organized as follows: section 2 presents the challenges presented on low resource language and the need for developing newly robust models, then we talk about the previous works on Pretraining transformers models on textual data, section 3 we describe our new DziriBERT model along with the novel scraped Algerian dataset. Section 4 presents the experiments and evaluation steps of the DziriBERT model. In section 5, we conclude.

²The original language of the first inhabitants of the region.

5.2 DziriBERT: an Algerian Language Model

In this section, we provide a detailed description of the data collection and pre-training settings utilized in the development of DziriBERT.

5.2.1 Training Data

Since there is no available text dataset for the Algerian dialect, we collected 1.2 Million tweets using Twitter API ³ that were posted from major and populated Algerian cities, using a set of popular keywords in the Algerian spoken dialect, such as Kho jeng: brother_ç, *arwah* jeng: come_ç, *jibli* jeng: get me_ç and الزوالي (eng: poor). The location was our only criterion to collect Algerian tweets without considering the used alphabet in their contents (which may be Arabic, Latin or a combination of both). We also collected a large amount of data from Facebook and youtube posts and comments. The final dataset after removing all duplicates and entries with less than three tokens contained 3.3 Million tweets (20 Million tokens), which represents almost 450 MB of text data. Then, we performed a light preprocessing on the collected data by (i) replacing all user mentions with *@user*; (ii) all email addresses with *mail@email.com*; and (iii) all hyperlinks with *https://anonymizedlink.com*. Finally, we randomly separate the collected data to a training set (having 1 Million entries) and a test set (having 100 Thousand entries).

The collected dataset is smaller in size when compared to other large-scale studies (Devlin et al. 2019; Antoun et al. 2020). However, it has been reported that we may need much less data than what we usually use when pre-training language models (Martin et al. 2020). The authors have shown that their official model (CamemBERT) trained on 138 GB performs similarly to another version trained only on a sample dataset of 4 GB. Here, we try to push this limit even further.

5.2.2 Language Modeling

DziriBERT uses the same architecture of BERT_{Base} (12 encoders, 12 attention heads, and a hidden dimension of 768). First, we train a WordPiece Tokenizer (Y. Wu et al. 2016) on our training data with a vocabulary size of 50 Thousand entries. Then, we train our language model using the Masked Language Modeling (MLM) task. Indeed, several studies have shown that the Next Sentence Prediction (NSP) task, originally used in BERT, does not improve the results of

³<https://developer.twitter.com/en/docs>

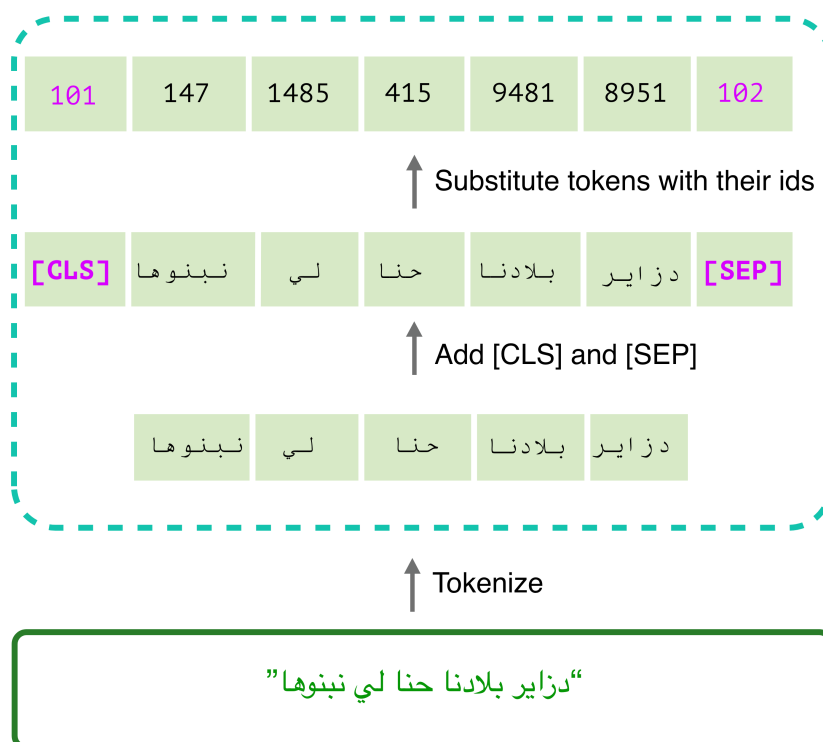


Figure 5.1: DziriBERT Tokenization process.

downstream tasks (Y. Liu et al. [2019](#); Lan et al. [2019](#)).

The tokenization process in DziriBERT is illustrated in Fig. [5.1](#), where each input sequence is typically formatted by adding a [CLS] token at the beginning and a [SEP] token at the end. The [SEP] token is used to separate multiple sentences or segments within a single input.

Since tweets have a short length, we used an MLM probability of 25% (instead of the usual 15%). We also set a batch size of 64 due to the limitations of our computational resources. The model has been trained on an AWS g4dn.2xlarge instance⁴ with 32 GB of memory and 1 NVIDIA T4 GPU. The training took almost 10 days to complete 50 epochs across the whole training set (around 800k steps). The final model created using PyTorch has been uploaded on the Transformers Hub to facilitate its use⁵.

5.3 Evaluation of DziriBERT

In order to compare DziriBERT with existing models, we have to fine-tune them on downstream tasks. It should also be noted that most related studies (Imane Guellil, Adeel, et al. [2021](#)) used

⁴<https://aws.amazon.com/ec2/instance-types/g4/>

⁵<https://huggingface.co/alger-ia/dziribert>

either non-publicly available datasets or contained only a small part of the Algerian dialect, which restricted the scale of potential comparative analysis. In this chapter, we considered two publicly available corpora that covered both Arabic and Roman scripts: Twifil (Moudjari, Akli-Astouati, and Benamara [2020](#)) and Narabizi (Touileb and Barnes [2021](#)).

5.3.1 Twifil

(Moudjari, Akli-Astouati, and Benamara [2020](#)) collected and annotated thousands of Algerian tweets according to the expressed sentiments and emotions. Most of them were written with Arabic letters but there were also many tweets written using the Roman scripts. The authors shared two publicly available⁶ datasets:

- Twifil sentiment: 9437 tweets annotated according to 3 polarity classes (positive, negative and neutral);
- Twifil emotion: 5110 tweets annotated according to the 10 Plutchik emotion classes (Plutchik [1984](#)).

5.3.2 Narabizi

The Narabizi corpus, originally published in (Djamé Seddah et al. [2020](#)), contains Algerian Arabic sentences written exclusively with the Roman script (Arabizi). In this chapter, we use the sentiment and topic classification datasets annotated in (Touileb and Barnes [2021](#)):

- Narabizi sentiment: 1279 sentences annotated according to 4 sentiment classes (positive, negative, mix and neutral);
- Narabizi topic: 1279 sentences annotated according to 5 topic classes (sports, societal, politics, religion and none).

These four datasets were used to compare DziriBERT with the two most known multilingual transformers (mBERT and XLM-R), and with all standard and dialectal Arabic models listed in Section [2.5.5](#) (AraBERT, QARiB, CamelBERT and MARBERT). Among the four available versions of CamelBERT, we evaluated the dialectal version (CamelBERT-da) and the one that has been pre-trained on a mix of all datasets (CamelBERT-mix).

Following (Devlin et al. [2019](#)), we used the final hidden state of the classification token ([CLS])

⁶https://github.com/kinmokusu/oea_algd

as a sentence representation followed by a one linear layer as a classifier. All models have been fine-tuned for three epochs using the Trainer Class of the Transformers library (Wolf et al. 2020) with its default settings. Ten different runs have been conducted for each model on each dataset according to the same 10 seeds that have been randomly generated. Still, the presented results may be reproduced using the shared Github repository [7](#).

5.3.3 Twifil

Tables [5.1](#), [5.2](#), [5.3](#) and [5.4](#) present the obtained results on the Twifil and Narabizi datasets. We calculate the accuracy and the macro averaged precision, recall and F1 score for each model on each dataset.

Table 5.1: Accuracy and macro averaged Precision, Recall and F1 score obtained by each model on the Twifil sentiment dataset.

| Model | Acc. | F1 | Pre. | Rec. |
|----------------|-------------|-------------|-------------|-------------|
| mBERT | 74.2 | 73.8 | 75.2 | 73.0 |
| XLM-R | 79.9 | 79.5 | 80.9 | 79.1 |
| AraBERT | 73.8 | 73.2 | 74.9 | 72.3 |
| QARiB | 78.8 | 78.2 | 79.0 | 77.9 |
| Camel-BERT-da | 75.2 | 74.6 | 76.0 | 74.0 |
| Camel-BERT-mix | 77.7 | 72.2 | 78.6 | 76.7 |
| MARBERT | 80.6 | 79.9 | 80.7 | 79.6 |
| DziriBERT | 80.5 | 80.0 | 81.1 | 79.5 |

Table 5.2: Accuracy and macro averaged Precision, Recall and F1 score obtained by each model on the Twifil emotion dataset.

| Model | Acc. | F1 | Pre. | Rec. |
|----------------|-------------|-------------|-------------|-------------|
| mBERT | 62.0 | 26.0 | 33.3 | 27.0 |
| XLM-R | 64.9 | 26.1 | 26.5 | 28.1 |
| AraBERT | 64.6 | 30.3 | 38.0 | 30.7 |
| QARiB | 68.9 | 39.2 | 42.2 | 38.7 |
| Camel-BERT-da | 66.0 | 34.6 | 38.7 | 34.6 |
| Camel-BERT-mix | 69.1 | 38.2 | 43.8 | 37.5 |
| MARBERT | 70.2 | 39.1 | 41.7 | 39.4 |
| DziriBERT | 70.4 | 40.1 | 42.8 | 39.6 |

5.4 Discussion

As shown in Tables [5.1](#) and [5.2](#), DziriBERT and MARBERT achieved the best results on the Twifil datasets (which are mainly composed of Arabic script). These two models, which are

⁷<https://github.com/alger-ia/dziribert>

Table 5.3: Accuracy and macro averaged Precision, Recall and F1 score obtained by each model on the Narabizi sentiment dataset.

| Model | Acc. | F1 | Pre. | Rec. |
|------------------|-------------|-------------|-------------|-------------|
| mBERT | 52.6 | 49.3 | 50.5 | 49.5 |
| XLM-R | 41.9 | 32.2 | 38.1 | 38.3 |
| AraBERT | 49.1 | 46.0 | 47.9 | 47.7 |
| QARiB | 55.0 | 52.9 | 53.7 | 53.4 |
| Camel-BERT-da | 40.9 | 35.5 | 36.0 | 40.1 |
| Camel-BERT-mix | 49.4 | 48.3 | 49.4 | 49.6 |
| MARBERT | 58.0 | 55.5 | 56.3 | 55.7 |
| DziriBERT | 63.5 | 61.2 | 62.0 | 61.4 |

Table 5.4: Accuracy and macro averaged Precision, Recall and F1 score obtained by each model on the Narabizi topic dataset.

| Model | Acc. | F1 | Pre. | Rec. |
|------------------|-------------|-------------|-------------|-------------|
| mBERT | 49.3 | 30.8 | 33.8 | 34.1 |
| XLM-R | 43.6 | 21.4 | 19.3 | 27.2 |
| AraBERT | 42.8 | 20.8 | 19.4 | 26.5 |
| QARiB | 45.7 | 29.7 | 29.9 | 32.4 |
| Camel-BERT-da | 43.7 | 21.5 | 20.2 | 27.3 |
| Camel-BERT-mix | 47.0 | 27.5 | 25.8 | 31.4 |
| MARBERT | 49.0 | 31.0 | 29.9 | 34.1 |
| DziriBERT | 62.8 | 54.8 | 64.0 | 53.2 |

both pre-trained on tweets, yielded better results than all other multilingual, standard Arabic, and dialectal Arabic models. However, DziriBERT yielded much better results on the Narabizi datasets (which are exclusively composed of Roman script) as shown in Tables 5.3 and 5.4. MARBERT is again in the second position but the difference with DziriBERT is much more important (+5.5% in sentiment accuracy and +13.8% in topic classification).

An error analysis step revealed that the Twifil datasets contain several entries that are not written in Algerian Arabic. DziriBERT tends to fail more often than MARBERT on documents that are written in standard Arabic or in other Arabic dialects, which may also explain the good results obtained by MARBERT on Twifil. Overall, our experiments have shown that DziriBERT can yield very good results despite the size of its pre-training dataset. For example, MARBERT has been trained on 128 GB of text (almost x1000 times larger than our pre-training corpus). Our experiments confirm that pre-training a dedicated model for the Algerian dialect on a small training set may give better results than pre-training a multi-dialectal model on a large-scale data. In fact, MARBERT has been trained on 128 GB of text (almost x1000 times larger than our pre-training corpus). However, it has been trained all Arabic dialects which

can be separated into +20 different groups, that vary from one region / country to another (Kareem Darwish et al. 2020). For example, the same sentence "عندي شقيقة" means "I have a headache" in the Algerian dialect, but is translated to "I have a sister" in standard Arabic and Middle-East dialects. Therefore, The same sentence may have different meanings according to the dialect spoken in each region. Therefore, we believe that the same sentence should have different representations according to the considered dialect. Still, DziriBERT is at least as good as MARBERT on the Algerian dialect and even much better when dealing with Roman characters.

Consequently, DziriBERT is not only the best performing model on the considered datasets, it is also the smallest one with less than 500 MB on disk (which should facilitate its deployment on Public Cloud Platforms).⁸

Furthermore, pre-training a mono-dialectal model allows us to safely reduce the number of entries in the vocabulary and therefore reduce the total number of parameters. Furthermore, DziriBERT's vocabulary contains a relatively small number of tokens when compared to the other baselines. Since the embedding layer concentrates most of the model parameters (Abdaoui, Pradel, et al. 2020), reducing the number of tokens should have a significant impact on the final model size (which should facilitate its deployment on Public Cloud Platforms). Table 5.5 presents the vocabulary length, the total number of parameters, and the final size on disk for all models studied here. As expected, even if all models share the same architecture (12 encoders, 12 attention heads, and 768 hidden dimensions), the total number of parameters varies from 110 Million to 278 Million. With its 50k vocabulary, DziriBERT is therefore one of the smallest models studied here.

5.5 DziriBERT for Named Entity Recognition

Named Entity Recognition (NER) is a crucial component in various natural language processing tasks, including event detection, user profiling, and information extraction, and it has been extensively employed across diverse languages. However, obtaining annotated data for low-resource languages or text corpora remains a scarce and challenging endeavor.

Recently, the introduction of the DziriBERT model has contributed significantly to the ad-

⁸Most Cloud Platforms limit the model size to 500 MB for serverless deployments

| Model | Vocab. | #Params (Million) | Size (MB) |
|----------------|--------|----------------------|--------------|
| mBERT | 106k | 167 | 672 |
| XLM-R | 250k | 278 | 1147 |
| AraBERT | 64k | 135 | 543 |
| QARiB | 64k | 135 | 543 |
| Camel-BERT-da | 30k | 110 | 439 |
| Camel-BERT-mix | 30k | 110 | 439 |
| MARBERT | 100k | 163 | 654 |
| DziriBERT | 50k | 124 | 498 |

Table 5.5: Models comparison according to the vocabulary length, the total number of parameters and the final size on disk.

vancement of NER research in the Algerian dialect. Numerous studies have documented the consistent performance of DziriBERT in identifying named entities within this particular dialect, demonstrating the model’s effectiveness and applicability in addressing the challenges inherent to low-resource languages (Touileb [2022](#); A. H. Dahou et al. [2023b](#); A. H. Dahou et al. [2023a](#)).

In the comparative study carried out by (A. H. Dahou et al. [2023b](#)), the performance of five BERT models, including AraBERT, MARBERT, ARBERT, mBERT, and DziriBERT, was assessed for Named Entity Recognition (NER) tasks in the Algerian dialect. The authors developed the DzNER dataset, an Algerian dialect NER dataset, annotated by two native Arabic speaker annotators who followed the same tagging guidelines. The dataset was compiled from two sources: the Algerian Dialect Corpus (Touileb and Barnes [2021](#)), which was built upon the NArabizi treebank (Djamé Seddah et al. [2020](#)), and data collected from Algerian Facebook pages. The latter source comprised 1,462 phrases and 15,868 tokens, automatically extracted from comments on Algerian Facebook pages.

In this analysis, we focus on comparing the performance of the DziriBERT model to that of other Arabic-based models, specifically AraBERT v0.2-T, AraBERTv2, MARBERT, and ARBERT. These models have been pretrained on a corpus that is 100 times larger than the one used for DziriBERT. Table [5.6](#) presents the performance of DziriBERT on the DzNER dataset, which contains Latin characters.

Specifically, in terms of precision (P), DziriBERT outperforms both AraBERT v0.2 and AraBERT v0.2-T, with only a slightly lower score than ARBERT. This highlights DziriBERT’s capability to accurately identify and classify entities in the Algerian dialect. Moreover, DziriBERT’s recall

(R) surpasses those of the other BERT models, indicating its effectiveness in detecting relevant entities without missing many true positives.

Finally, the F1 score, which represents the harmonic mean of precision and recall, demonstrates that DziriBERT achieves a balanced performance between these two metrics. Although its F1 score is slightly lower than that of AraBERT v0.2, it outperforms AraBERT v0.2-T, ARBERT, and MARBERT. This suggests that DziriBERT is well-suited for handling the intricacies of the Algerian dialect and can serve as a strong choice for NER tasks in this specific context.

Table 5.6: Metrics Comparison for Different Arabic BERT Models on DzNER (A. H. Dahou et al. 2023b)

| Metrics | AraBERT v0.2 | AraBERT v0.2-T | ARBERT | MARBERT | DziriBERT |
|---------|--------------|----------------|--------|---------|-----------|
| P | 0.692 | 0.663 | 0.720 | 0.670 | 0.680 |
| R | 0.616 | 0.566 | 0.529 | 0.520 | 0.593 |
| F1 | 0.652 | 0.610 | 0.610 | 0.586 | 0.634 |

NERDz (Touileb 2022), is an extension of the NArabizi treebank (Touileb and Barnes 2021), and is annotated using the IOB2 scheme for eight entity types: *PER*, *GPE*, *ORG*, *NORP*, *EVT*, *LOC*, *PROD*, and *MISC*. NERDz comprises 1,566 annotated entities, from which 1,229 are in train, and 180 and 157 are in dev and test respectively. A study by (Touileb 2022) investigated the performance of DziriBERT on NERDz, a NER corpus of Algerian dialects written on multiple scenarios, Algerian Arabic, Code-switching, and NArabizi.

Table 5.7: Strict entity type-level F1-score for the four most frequent entity types for the three scripts NArabizi (NA), Alg-Arabic (Ar), and codeswitched (CS) in test (Touileb 2022)

| | NA | Ar | CS |
|------|-------|-------|-------|
| PER | 46.15 | 63.55 | 51.35 |
| GPE | 53.54 | 66.66 | 45.51 |
| ORG | 30.43 | 28.16 | 26.54 |
| NORP | 23.37 | 47.05 | 26.41 |

In Table 5.7, the performance of DziriBERT is evaluated in terms of strict entity type-level F1-score for the four most frequent entity types in three different scripts: NArabizi (NA), Alg-Arabic (Ar), and codeswitched (CS) (Touileb 2022). The table highlights the differences in model performance across various entity types and scripts.

The results indicate that DziriBERT performs best on the Alg-Arabic (Ar) script when identifying the PER (person) and GPE (geopolitical entity) entity types, with F1-scores of 63.55 and 66.66, respectively. For the NArabizi (NA) script, the model achieves the highest performance for the GPE entity type with an F1-score of 53.54, while the lowest performance is observed for the NORP (nationality, religious, or political group) entity type with an F1-score of 23.37.

When handling codeswitched (CS) text, DziriBERT shows the best performance on the PER entity type with an F1-score of 51.35, while it performs relatively poorly on the ORG (organization) and NORP entity types with F1-scores of 26.54 and 26.41, respectively.

Overall, table 5.7 provides valuable insights into the strengths and weaknesses of the DziriBERT model when applied to different entity types and scripts. The varying performance across entity types and scripts highlights the importance of considering both factors when developing and evaluating named entity recognition models for multilingual and codeswitched text.

5.6 Limitations

While pre-training a mono-dialectal model may give better results and produce smaller models, there are also important advantages to rely on multidialectal models. This includes the ability to do cross-dialectal transfer when the labeled task data is only available for a small number of dialects. We believe that choosing between mono-dialectal and multidialectal models is similar to choosing between monolingual and multilingual ones and that the decision should be made according to each specific need. However, we must note that Arabic dialects are similar to each other (at least partially). Therefore, Arabic dialects should help each other when building multidialectal models and the zero-shot cross-dialectal transfer should give decent results. That being said, we still believe that DziriBERT fills a gap left by the current models in handling Algerian text contents, especially when written with the Roman script.

The pre-training data used in this project comes from social media. Therefore, the Masked Language Modeling objective may predict offensive words in some situations. Modeling these kinds of words may be either an advantage (e.g. when training a hate speech model) or a disadvantage (e.g. when generating answers that are directly sent to the end-user). Depending on your downstream task, developers may need to filter out such words, especially when returning automatically generated text to the end user.

5.7 Conclusion

The present chapter aims to elaborate on the pre-training and evaluation of DziriBERT, a language model designed to address the challenges posed by the Algerian Dialect. In light of the increasing interest in natural language processing tasks for non-standard languages, such as Algerian Arabic, the need for appropriate models that are capable of capturing the specificities and complexities of such languages has become pressing. Our research efforts have thus been devoted to developing a model that can effectively handle the peculiarities of the Algerian dialect, while outperforming existing models even when pre-trained on a comparatively smaller corpus.

The purpose of this study is to promote the development of NLP applications for low-resource dialects, particularly the Algerian Dialect. To facilitate this goal, we have made our pre-trained DziriBERT model publicly available to the community, along with pre-trained versions for sentiment, emotion, and topic classification tasks. Our objective is to provide a valuable resource to researchers and practitioners working in this field, with the aim of encouraging the advancement of NLP capabilities for under-resourced dialects such as the Algerian Dialect.

In addition to sharing our pre-trained model, we suggest future research endeavors in compiling and annotating more Algerian datasets for other NLP tasks, such as Named Entity Recognition and Question Answering. These tasks are particularly relevant for practical applications in industries such as e-commerce, social media, and news media, where effective processing of named entities and accurate answering of questions is essential for improving user experience and engagement.

Chapter 6

Effective approaches for Arabic Text Classification

Text classification emerged as one of the main applications of natural language processing tasks such as topical analysis, sentiment analysis, and news classification where several deep learning models, as well as multi-classifier systems, have been put forward. This challenge is even more stressed when using low-resource languages. In this study, we present a detailed study of the most recent advanced neural network models and pretrained language models applied for Arabic news classification. We evaluated and compared the performance of 8 neural-network-based models and proposed a new TransConvNet architecture that outperforms the most efficient neural-network models. Additionally, we fine-tuned 5 large pretrained multi-lingual and Arabic language models in both Multi-class and Multi-label text classification tasks. where the best classifier achieves accuracy scores ranging from 93% to 98.55% on six popular Arabic text classification benchmarks. Besides, a new large collected dataset that consists of 123,408 short Arabic news headlines (ANHD) has also been released to the research community.

6.1 Introduction

In the era of Big Data and the democratization of communication tools, the size of electronic documents has continued to increase at an unprecedented scale. Text classification, as the task of assigning correct labels to text documents, becomes of paramount importance. This importance is emphasized by the scope and the growing number of critical tasks that depend

on the result of this classification. This includes, for example, news categorization (Einea et al. [2019](#)), spam filtering (Alsmadi et al. [2019](#)), emotion and sentiment recognition from user comments or reviews (Farha et al. [2021b](#); Al-Dabet, Tedmori, and Al-Smadi [2021](#)), online abuse/hate speech detection in social media platforms (R. Ali et al. [2022](#); Berrimi, Abdelouaheb Moussaoui, Oussalah, et al. [2020b](#)), among others.

Traditionally, text classification is seen as a machine learning (ML) problem where documents are represented using their corresponding numerical feature vectors, e.g., Tf-idf, which are then fed to a given classifier (e.g., support vector machine, linear regression) that utilizes some training sample to build the classifier model, which is, in turn, employed to predict the class label of the inputted text document (See, e.g., (Feldman et al. [2006](#))). Since the last decade, Deep Learning (DL) approaches have widely been acknowledged as the mainstream technology in text classification task, and gradually substituted to traditional machine learning techniques.

Nevertheless, significant challenges still exist to devise efficient deep learning classification systems due to several inherent limitations observed in the state-of-the-art DL models. First, there is a conceptual difference among document-label mappings where one distinguishes cases where there is a sound intuitive mapping between a given document to a given category, while in other cases, a document is rather genuinely found to be associated with more than one category (multi-labels) (T. Wang et al. [2011](#)). Indeed, often, long document bears multiple topics and contexts, which renders the task of identifying a dominant overall context rather difficult to impossible (Deng et al. [2021](#)).

This clearly questions the quality of the training samples employed by DL models where each document is uniquely assigned to one unique label. Second, the growing scale and size of the unstructured generated textual data are becoming increasingly important; hence, extracting relevant information from long and large documents becomes challenging and time-consuming. Third, often, there is wide discrepancy in terms of documents employed in the training set and those employed in testing phase, which challenges the generalization capability of any classifier model. This discrepancy arises from the difference in document size, topic-focus, vocabulary size, linguistics cues, style, proportion of noise, among others.

News documents are a particular case of electronic documents that are originated from online news repositories or platforms that deliver updates about contemporary events or activities.

News headlines are critical parts of the news articles written in a specific lexical choice (Napu 2018; Schulz 2008) that recapitulates the content of the (possibly long) news article.

Additionally, often, news headline attracts the reader's attention to whether to continue reading the whole article or not, due to its acknowledged role in shaping the user's opinion about the document content. Effective classification / categorization of news-headline documents would assist the readers in a better recommendation according to their preferences and profiles. Nevertheless, the growing number of sources and platforms renders the identification of number and type of news category very difficult, especially with the increasing multilingual and noise content of the news documents as well as data sparsity issues (Alsmadi et al. 2019), which calls for further research on this issue.

This chapter focuses on classification of *news headlines* and *news* documents in Arabic language. Especially, handling Arabic language bears extra difficulty due to its high orthographic ambiguity level (Kareem Darwish et al. 2020), rich morphology, (N. Habash 2019) dialectal variation (Berrimi, Abdelouahab Moussaoui, et al. 2020), and the lack of reliable NLP resources (Kareem Darwish et al. 2020).

Multiple studies have surveyed recent research applications in Arabic text classification such as (El Rifai et al. 2021; Alzawaydeh et al. 2018; Mohammed et al. 2019; Elsayed et al. 2020). Although these studies used classical ML models as well as deep models like LSTM, CNN, or BiLSTM models, recent transformer architectures have not yet been investigated in the multi-class and multi-label Arabic text classification tasks, and lack comparison with state-of-the-art methods. For instance, it is still unknown which model best suits the long and short text document classification paradigm and whether fine-tuning latest transformers-based models or pretrained embedding models such as FastText (Tomas Mikolov, Grave, et al. 2018), word2vec can outperform other neural net-based models.

In this chapter, we are interested in applying recent breakthroughs in deep learning in both domains: neural network research and word representation research to Arabic text classification tasks involving both short and long news articles. The objectives of this chapter are the following:

- Comprehend the difference between long and short-length text document classification problems for both multi-class and multi-label tasks in Arabic language.

- Develop a new deep-learning model using most recent transformer architectures and test its performance with respect to most effective baselines and pre-trained word embedding models on long and short text datasets for both multi-class and multi-label text classification tasks in Arabic language.
- Contribute to Arabic NLP community by new publicly available resources.

To achieve the aforementioned objectives, the following contributions have been made:

- We surveyed the state-of-the-art methods for Arabic text classification and presented a concise literature review on the issue.
- We replicated and evaluated 13 DL models proposed for text classification on Arabic news documents and news headlines classification.
- We proposed a new classification model based on the transformer model architecture that outperforms previous SOTA neural networks-based models on (6) six benchmark datasets.
- We evaluated Arabic BERT-based models for Arabic text classification on multi-class and multi-label domains on SANAD (a multi-class long-sentence news) dataset, NADIA (a multi-label long-sentence news) dataset (Einea et al. 2019) and ANHD (a short multi-class) dataset. Finally, we show the effectiveness of the pretrained contextual embeddings.
- We presented a new publicly available dataset of Arabic news headlines, called *ANHD*, which contains 123,408 short headlines grouped into six balanced classes: technology, health, politics, culture, sports, and economics.
- We studied different variations of the Transformer model architecture and compared their performances to classical neural network models and pretrained contextual embeddings.

The research results present a promising breakthrough in deep learning applied to Arabic text classification (ATC) and also have a direct impact on future research in ATC and in Arabic NLP in general.

Section 2 presents the research methodology as well as the datasets for further training and validating the models and finally the preprocessing phase.

Section 4 presents the experimental part of our research where several deep learning architectures and fine-tuned Bert-based models are discussed.

In Section 5, we report the classification performances of our experiments and then discuss the results accordingly when using the various datasets.

Finally, Section 6 summarizes the research findings and concludes the effectiveness of the recent DL models in Arabic Text classification.

6.2 Methodology

In this section, we describe the research methodology used to achieve our research objectives.

This research can be divided into two main directions:

1. Replicating, proposing, and training deep neural network-based models such as RNN-based models, CNNs, and the transformer model architecture, as illustrated in Fig. 6.1.
2. Fine-tuning recent pretrained language models such as mBERT, AraBERT, and MARABERT. The goal of this part of the research is to compare the performance of recent neural network models using FastText embeddings with the fine-tuning of BERT-like models. as illustrated in Fig. 6.3.

By following this methodology, we aim to provide a comprehensive comparison of different approaches to text classification and to identify the strengths and limitations of each approach. This will help us to understand the performance of different models on a variety of experiments and datasets, and to identify the most effective approaches for different types of natural language processing applications.

6.2.1 Neural-networks-based approaches

In this section, we summarize our deep neural networks-based approaches that have been proposed in previous works. For these models' architecture, the input embedding layer has FastText embeddings (Tomas Mikolov, Grave, et al. 2018) weights. The models that we have considered include LSTM, GRU, Bidirectional RNN, CNN and hybrid RNN-CNN models. All these models have been previously discussed in chapter 3.

6.2.2 Proposed TransConvNet model

Due to the complex nature of the Arabic language, it is challenging to extract important features from input sequences, particularly when dealing with long text documents. This is a problem

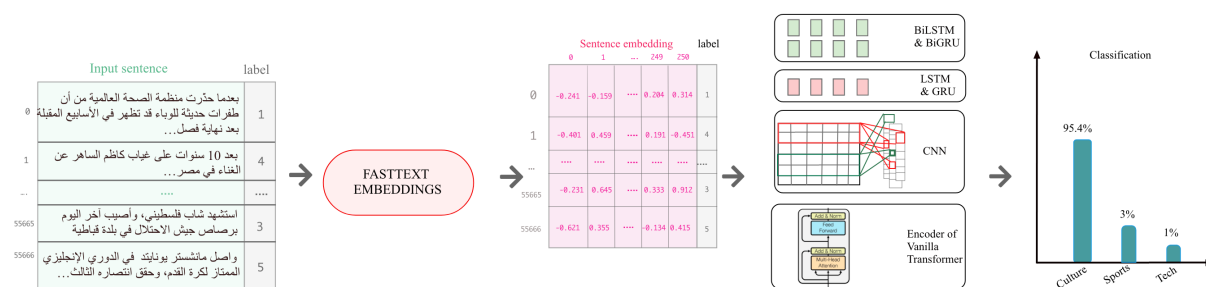


Figure 6.1: Overall architecture of neutral-networks-based models experiments

that is commonly encountered in Arabic NLP tasks, such as those involving the SANAD and NADIA datasets. To address this challenge, we propose a new classification model that is designed to better handle the complexity of features in text documents in Arabic. Our model is composed of the following components:”

Our proposed model is built by stacking encoders from the vanilla transformer. To determine the number of encoders, we conduct multiple experiments and use the results from these experiments to inform our decision, as shown in table 6.4. The output vectors from the encoders are fed into a 1-dimensional convolutional neural network (CNN) layer that consists of 128 filters and uses *relu* activation functions. This layer helps extract important hidden features from the vectorized tokens. Our model combines the flexibility of the attention operations used in the encoder level with the deep processing capabilities of the CNN building block, providing improved performance over previous models.

Let X be a set of feature vectors of input tokens obtained from the Multi-head attention layers of the encoder section of the transformer model. These feature vectors can be combined using the following equation:

$$X_{1:n} = x_1 \oplus x_2 \oplus \dots \oplus x_n \quad (6.1)$$

where \oplus denotes vector concatenation.

The convolution operations at this level use a set of filters \mathbf{k} (128 in this case) to extract and produce relevant features from a window of input features w . The resulting feature vectors can be calculated using the following equation:

$$F_i = k(w \bullet x_{i:i+h-1} + bias) \quad (6.2)$$

where h is the number of words convolved by the filter \mathbf{k} .

To output the final feature vector, the 1D-CNN layer sums all possible feature vectors from the convolved input vector (obtained by the set of operations performed by the Multi-headed attention layers). This can be expressed as follows:

$$F = [f_1, f_2, \dots, f_{n-h+1}] \quad (6.3)$$

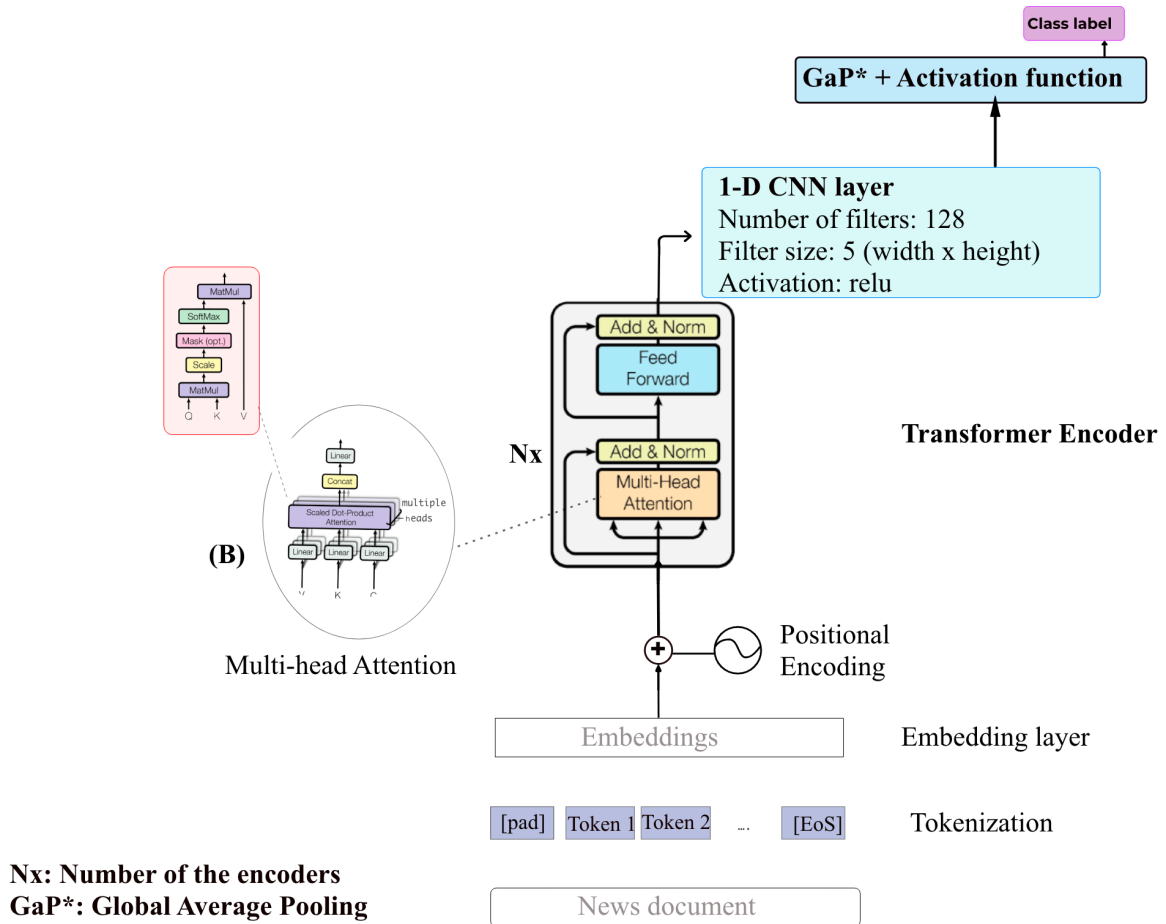


Figure 6.2: Our TransConvNet model architecture

Table 6.1: The hyperparameters of our proposed TransConvNet model

| Hyperparameter | Value |
|---------------------------|-----------|
| Number of encoders | 1,2,3,4,6 |
| Number of Attention heads | 2,4,6 |
| Number of filters | 128, 64 |

After the feature extraction process, the resulting feature vectors are passed as input to a

multilayer perceptron (MLP) neural network. The MLP consists of multiple hidden layers of neurons, each of which applies a rectified linear unit (ReLU) activation function to the weighted sum of its inputs. This allows the network to learn a non-linear mapping from the input feature vectors to the output classes or labels.

To prevent overfitting and improve the generalization performance of the MLP, a dropout layer is applied between each pair of hidden layers. The dropout layer randomly sets a fraction of the inputs to zero, forcing the network to learn redundant representations of the input data that can be used to make accurate predictions even when some of the input features are missing.

Finally, the output of the last hidden layer is passed through a final activation layer that applies a softmax function for multi-class classification tasks, or a sigmoid function for multilabel classification tasks. The output of the activation layer is a probability distribution over the possible classes or labels, which can be used to make predictions about the input data.

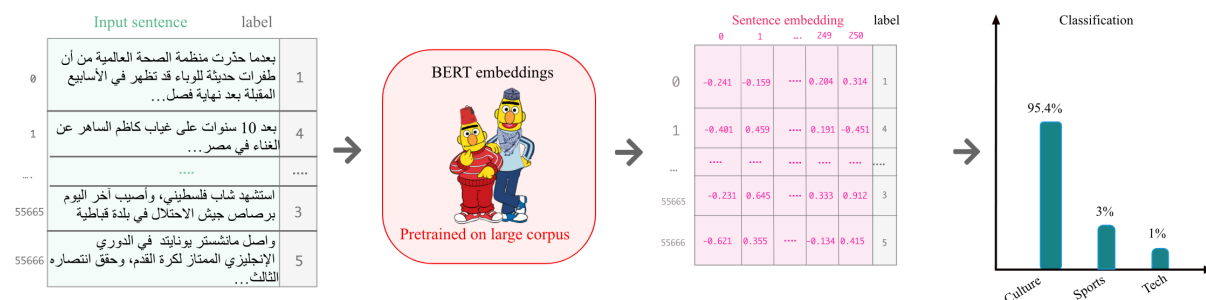


Figure 6.3: Pretrained language models experiments overall architecture

6.3 Experimental setup

In our experiments, we evaluate the performance of different deep learning models on six different classification datasets. We use these datasets to compare the performance of the models using different word embedding techniques and different hyperparameter settings. The experiments were performed on an NVIDIA A100 GPU with 40 GB of RAM. This high-end hardware allows us to run complex and computationally intensive experiments with ease.

6.3.1 Arabic text classification datasets

In order to assess the effectiveness of various deep learning models and word embeddings, we utilized the SANAD and NADiA datasets, as previously mentioned in Section [2.5.6](#). Furthermore, we also introduced two new datasets to complement our evaluation.

Arabic News headlines dataset

We introduce a newly collected dataset for Arabic News Headlines (ANHD) in which a collection of 123,408 short headlines belonging to six news categories namely: Technology, Health, Politics, Culture, Sports, and Economics. The dataset was scraped from various Arabic news websites: Alarabia¹, Echorok online², aitnews³, bcc⁴, Ennahar⁵, al-ghad⁶ and masrawy⁷. The dataset is composed of 100k unique words, the average length of the headlines is 7 tokens, the maximum length is 15 tokens, and more than 40k news headlines have less than 7 tokens, as illustrated in Fig. 6.5

The ANHD dataset is made public for research use. Our dataset is well-defined and standardized, which makes it easy for other researchers to replicate our experiments and compare their results to ours. This can help ensure the robustness and reproducibility of our findings. Additionally, our dataset is the largest available dataset, to our knowledge, for short-text Arabic data, which makes it useful for a wide range of research applications in Arabic short-text classification.

Table 6.2 shows the number of headlines for each news category.

Table 6.2: Number of headlines per class in the ANHD dataset.

| Class | # of headlines |
|------------|----------------|
| Politics | 19610 |
| Medical | 20956 |
| Technology | 22856 |
| Culture | 19436 |
| Sports | 19851 |
| Economics | 20699 |

The collection process was the following: We selected multiple Arabic news websites and scraped all news headlines of each category by visiting each page on the website's sections (Technology, Medical, Politics, Culture, Sports, and Economics) from December 2020 until the very first page on each website. However, we noticed that some categories (Technology, Health, Culture) had fewer data, so we selected additional online news media that fit our selection criteria. We only

¹www.Alarabia.com

²www.echoroukonline.com

³www.aitnews.com

⁴www.bcc.com/arabic

⁵www.ennaharonline.com/

⁶www.alghad.tv/

⁷www.masrawy.com/

selected subsections of the already scraped sections to avoid duplication and redundancy. For example, in the Sports category, we collected news headlines for international sports news from Alarabia and premier league news headlines from Echoroukonline. We applied this process to the rest of the categories to ensure a balanced dataset.

SANAD-ALL

In this study, we performed long multi-class text classification experiments using the SANAD-ALL dataset, which is a revised and expanded version of the original SANAD dataset. The original SANAD dataset consists of three subsets of Arabic news collected from three online news portals: Alarabia⁸, Khalij⁹, and Akhbarona¹⁰. Our experiments used all three of these subsets, as well as the expanded SANAD-ALL dataset, which combines them into a single comprehensive dataset. The size and class distribution of each of these datasets are shown in Table 6.3 and Fig. 2.4, respectively. The average length of the news articles in the SANAD datasets is 260 tokens.

As shown in Fig. 6.4, the ANHD and SANAD-ALL datasets are the largest in size. Additionally, the imbalanced class distribution in the Akhbarona and Arabia datasets is evident, which could potentially impact the performance of deep learning models trained on these datasets.

Table 6.3: Number of headlines in each dataset for training, validation, and total.

| | SANAD | NADIA | ANHD | Alarabia | Khalij | Akhbarona |
|-------------------|--------------|--------------|-------------|-----------------|---------------|------------------|
| Train | 144,921 | 28,332 | 112,355 | 15,754 | 36,400 | 56,724 |
| Validation | 36,231 | 7,084 | 11,053 | 6,3014 | 9,100 | 14,182 |
| Total | 181,152 | 35,416 | 123,408 | 78,768 | 45,500 | 70,906 |

6.3.2 Hyperparameter tuning

Table 6.4 summarizes the hyperparameters of the different models that we evaluated in our experiments. These hyperparameters include the number of layers, the number of parameters, and the pretraining tasks used to train the models. We carefully selected these values through a process of experimentation and optimization, in which we conducted multiple iterations to determine the settings that would lead to the best performance. As a result, the values in Table 6.4 represent the optimal settings that we found for each model.

⁸<https://www.Alarabia.com>

⁹<https://www.akhbar-alkhaleej.com>

¹⁰<https://www.akhbarona.com>

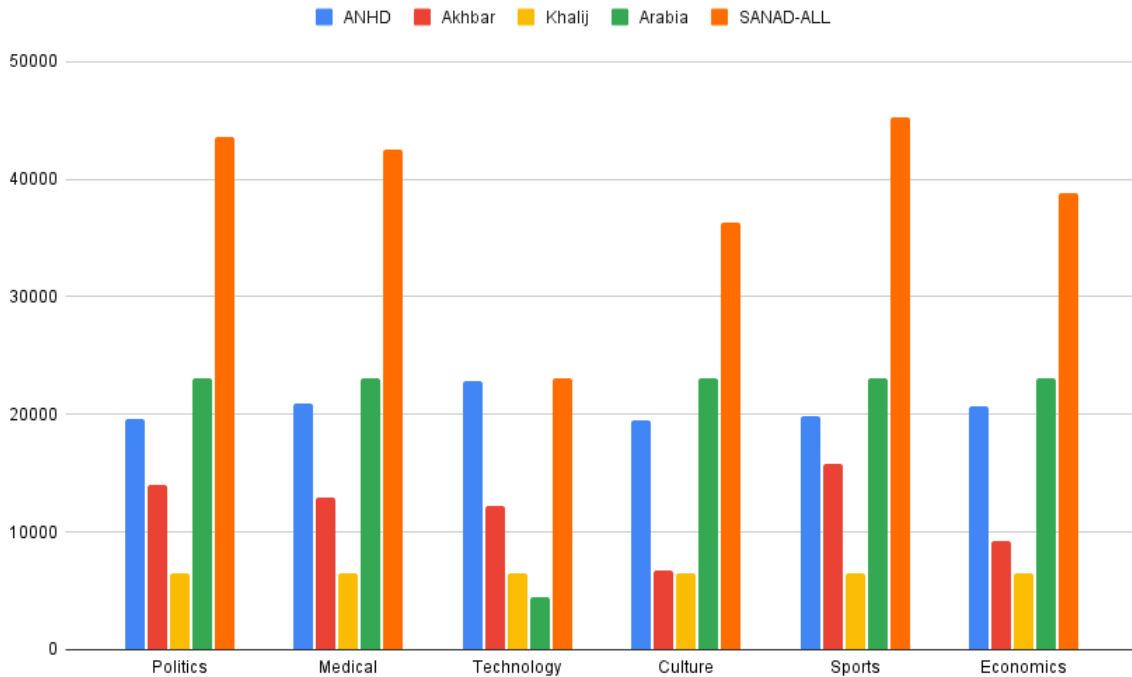


Figure 6.4: Class distribution of the multi-class datasets.

Table 6.4: Models' hyperparameters

| Classifier | Reccurent dropout | Hidden units | Dropout rate | N° filters | Filter size | Attention heads |
|---------------------|-------------------|--------------|--------------|------------|-------------|-----------------|
| LSTM | 0.5 | 250 | 0.5 | - | - | - |
| GRU | 0.5 | 250 | 0.5 | - | - | - |
| BiLSTM | 0.5 | 250 | 0.5 | - | - | - |
| BiGRU | 0.5 | 250 | 0.5 | - | - | - |
| C-BiLSTM | 0.5 | 250 | 0.5 | 128 | 5 | - |
| C-BiGRU | 0.5 | 250 | 0.5 | 128 | 5 | - |
| CNN | - | - | 0.1 | 128 | 5 | - |
| Vanilla-Transformer | - | - | 0.1 | - | - | 4 |
| Transf-CNN | - | - | 0.5 | 128 | 5 | 4 |

6.4 Results and Discussion

In this section, we report the experimental results of the aforementioned deep learning models. In order to determine which model is suitable for both multi-class and multi-label text classification tasks, we trained and fine-tuned a total of 13 different models on 6 classification datasets. The results of the experiments are shown in Tables 6.5 and 6.6, which include the values of accuracy, F1-score, and precision for the three datasets. The top values are in bold text.

As shown in the tables, the proposed TransConvNet model demonstrated the highest classification performance among the neural network models. The specific characteristic of our proposed model helped in the extraction of hidden features, leading to improved performance compared

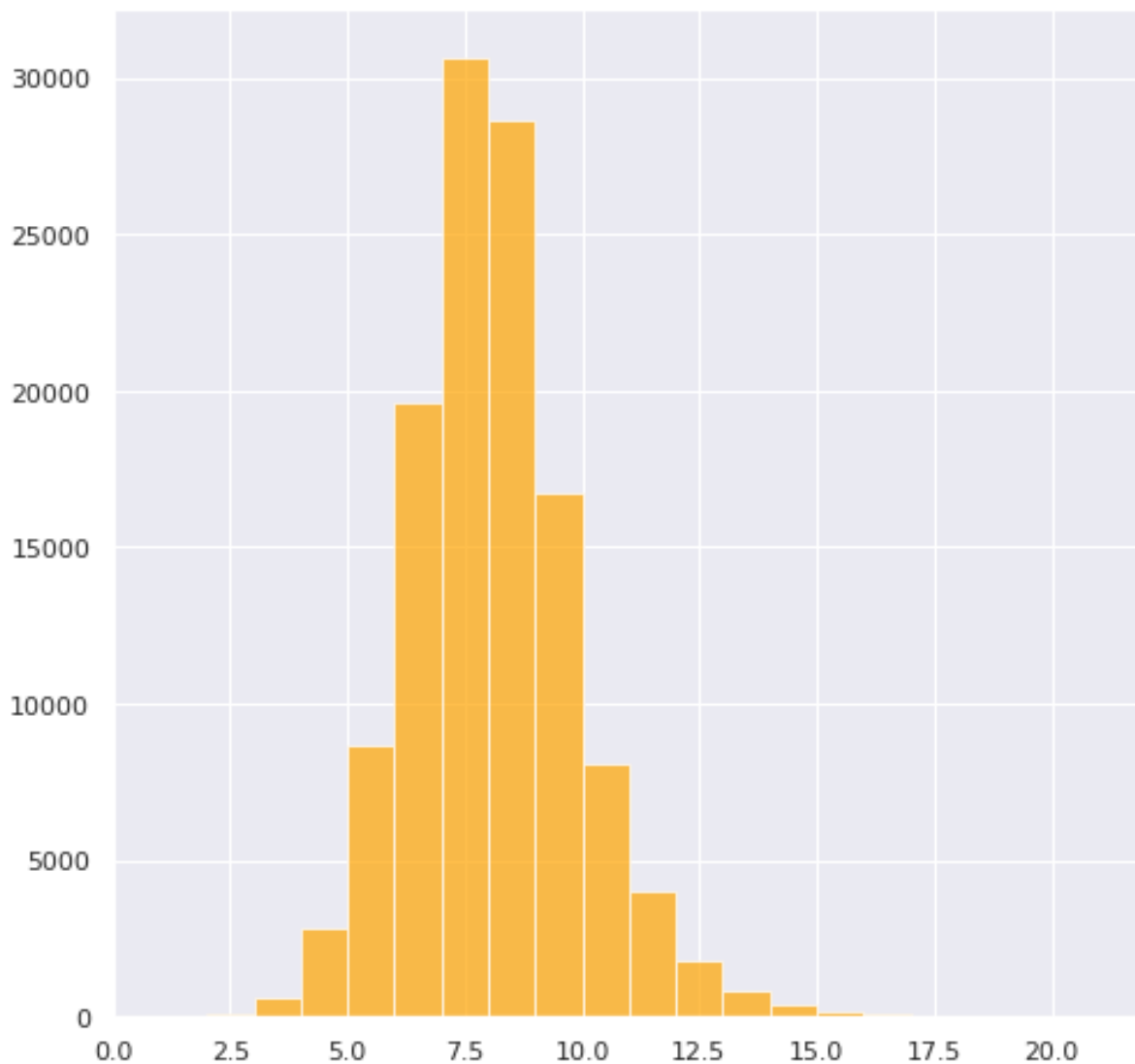


Figure 6.5: Headlines length (number of tokens per headline) distribution in the collected ANHD dataset.

Table 6.5: Classification Results on the ANHD, SANAD-ALL and NADIA datasets

| Model | ANHD | | | SANAD-ALL | | | NADIA | | |
|---------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | Acc | AvgF1 | AvgPrec | Acc | AvgF1 | AvgPrec | Acc | AvgF1 | AvgPrec |
| LSTM | 91.33 | 91.32 | 91.4 | 92.41 | 92.45 | 92.11 | 86.25 | 57.79 | 79.21 |
| GRU | 91.45 | 91.1 | 91.23 | 92.36 | 92.86 | 93.01 | 88.54 | 78.29 | 60.34 |
| BiLSTM | 91.95 | 92.01 | 92.98 | 93.33 | 93.00 | 93.09 | 89.07 | 66.10 | 82.12 |
| BiGRU | 91.79 | 91.96 | 93.03 | 92.91 | 93.04 | 93.07 | 88.78 | 57.20 | 79.54 |
| CNN | 92.02 | 92.07 | 93.53 | 93.25 | 93.17 | 93.10 | 83.84 | 64.14 | 78.41 |
| C-BiGRU | 92.67 | 92.34 | 93.26 | 93.43 | 93.48 | 93.76 | 86.05 | 55.55 | 76.13 |
| C-BiLSTM | 91.93 | 92.07 | 93.12 | 93.42 | 93.42 | 93.41 | 88.07 | 64.84 | 81.02 |
| Vanila-Transformer | 93.99 | 94.12 | 94.10 | 94.11 | 94.32 | 94.50 | 92.52 | 92.13 | 91.67 |
| TransConvNet | 94.27 | 94.41 | 94.75 | 94.29 | 94.51 | 94.72 | 92.74 | 92.37 | 92.64 |
| AraBERT | 97.72 | 93.5 | 93.63 | 97.10 | 97.21 | 97.09 | 93.28 | 93.32 | 93.17 |
| ARBERT | 97.97 | 98.20 | 98.38 | 96.74 | 97.41 | 97.02 | 93.53 | 93.14 | 93.43 |
| MARBERT | 97.10 | 97.21 | 97.54 | 97.38 | 97.18 | 97.39 | 93.41 | 93.06 | 93.12 |
| mBERT | 89.27 | 79.10 | 79.04 | 95.56 | 94.07 | 95.00 | 84.01 | 84.34 | 83.51 |

Table 6.6: Classification results on 3 subsets of SANAD

| | Arabia | | | akhbar | | | Khalij | | |
|---------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | Acc | AvgF1 | AvgPrec | Acc | AvgF1 | AvgPrec | Acc | AvgF1 | AvgPrec |
| LSTM | 96.47 | 96.55 | 97.14 | 91.29 | 91.34 | 91.41 | 93.41 | 93.45 | 94.09 |
| GRU | 96.52 | 96.53 | 96.82 | 91.45 | 91.33 | 91.11 | 92.85 | 92.86 | 92.86 |
| BiLSTM | 96.57 | 96.60 | 97.16 | 91.05 | 91.18 | 91.19 | 93.38 | 93.41 | 94.09 |
| BiGRU | 96.77 | 96.79 | 96.99 | 90.95 | 90.98 | 91.01 | 92.91 | 92.93 | 93.71 |
| CNN | 96.41 | 96.37 | 97.11 | 91.35 | 92.02 | 92.18 | 93.25 | 93.24 | 94.09 |
| C-BiLSTM | 96.57 | 96.53 | 96.84 | 91.53 | 91.24 | 91.15 | 93.40 | 93.41 | 94.01 |
| C-BiGRU | 96.44 | 96.41 | 96.75 | 91.45 | 91.11 | 91.03 | 93.42 | 93.42 | 94.01 |
| Vanila-Transformer | 96.98 | 97.04 | 97.10 | 91.82 | 92.08 | 92.00 | 94.10 | 94.13 | 94.10 |
| TransConvNet | 97.19 | 97.32 | 97.27 | 92.14 | 92.19 | 92.61 | 96.55 | 96.23 | 96.45 |
| AraBERT | 98.48 | 98.35 | 98.59 | 95.40 | 95.19 | 95.31 | 98.58 | 98.09 | 98.13 |
| ARBERT | 98.55 | 98.12 | 98.10 | 97.04 | 96.42 | 97.27 | 98.79 | 98.11 | 98.13 |
| MARBERT | 98.05 | 96.10 | 97.04 | 95.17 | 95.12 | 96.19 | 98.15 | 98.36 | 98.10 |
| mBERT | 92.19 | 92.09 | 92.24 | 88.55 | 88.09 | 88.10 | 92.34 | 92.12 | 92.22 |

to other neural network models. These results indicate that the TransConvNet model is effective for this task and could be a valuable tool for future research in Arabic short-text and long-text classification.

However, all of the Arabic pretrained models outperformed the neural network models, with the pretrained ARBERT model achieving the best performance on all of the different datasets. This suggests that pretrained contextual word embeddings (BERT-based models) are the most effective approach for text classification tasks in Arabic. Furthermore, the more pretraining a model has, the better it performs when fine-tuned for downstream tasks. In particular, the use of context-dependent embeddings, such as those used in BERT-based models, tends to produce better results compared to FastText embeddings, which are not context-dependent. Additionally, Arabic BERT-based models trained exclusively on Arabic text corpora outperformed the larger mBERT model, which is trained on a mixture of languages, indicating that models trained on more specialized data can be more effective for specific tasks. The performance gap between the Arabic BERT model and the multilingual BERT model indicates that these pretrained language models yield better performance when they are trained on a large amount of data that is specific to the language they are being used for. This highlights the importance of having high-quality, domain-specific training data in order to achieve optimal performance with these models.

The proposed model’s training time was found to be longer for long-text classification experiments compared to headline classification tasks. This is due to the increased computational

demands of the Self-attention layer in the long-text classification setting. In a long-text classification experiment, the Self-attention layer needs to calculate the attention weights of each word in relation to the rest of the words in the sentence, which results in a larger number of calculations that need to be performed. This increase in computational complexity leads to a longer training time for the model. In contrast, in a headline classification experiment, the Self-attention layer only needs to calculate the attention weights for a smaller number of words, as the text is shorter and typically consists of a single sentence or phrase. As a result, the training time for the Transformer model is shorter in this case.

Fine-tuning large pretrained language models leads to better performance compared to neural network (NN) models, due to the amount of pretraining data available to the models. For example, the ARBERT model is pretrained on 100 GB of Arabic text, while the FastText model is pretrained on less than 1 GB of text. Additionally, the BERT-based models have a larger number of parameters (e.g. MARBERT has over 160M parameters) compared to the proposed TransConvNet model (3M parameters), which allows them to capture more information from the input data and improve their performance. Furthermore, the word-tokenization technique used in BERT-based models is able to cover unseen tokens in the training data and assign contextual vectors, which is difficult for other types of embeddings to do. This contributes to the improved performance of these models on Arabic text classification tasks.

Additionally, it is observed that all of the models perform poorly in multi-label experiments (MLE) compared to multi-class experiments. This is likely due to the fact that the ground-truth labels in MLE are often correlated and not equally unpredictable, which can make it difficult for classifiers to accurately predict the labels in these cases. Therefore, it is important to carefully consider the specific challenges and limitations of each classification task when selecting and training a model.

6.4.1 Statistical evaluation

We conducted a statistical test (t-test) to determine whether the superior performance of our model on the three datasets (SANAD, ANHD, and NADiA) was statistically significant. The null hypothesis of the test assumed that the means of the two populations (our model and the second-ranked model) were equal. By comparing the results of the two populations through randomization of the training samples, we found that the null hypothesis was rejected with a

$p < 0.0001$ in all cases, indicating that the superiority of our model was statistically significant.

Table 6.7: Statistical test on the performance of the proposed TransConvNet model.

| Dataset | ANHD | | SANAD | | NADiA | |
|-------------|--------|---------|--------|---------|--------|---------|
| | T-test | Pvalue | T-test | Pvalue | T-test | Pvalue |
| Test Result | 80.13 | < 0.001 | 85.63 | < 0.001 | 88.32 | < 0.001 |

6.4.2 Error Analysis

Despite the challenges imposed by short-text data, our proposed TransConvNet model achieved good results in accurately classifying headlines. However, we observed that some of the misclassified headlines were even difficult for the authors to determine the correct class. For example, in the culture headlines that were confused with economics, such as "(Organizing the Cultural Economy Forum next November)" and "(concerts to revive the economic scene in the country)", the model may have struggled to differentiate between these two classes. Additionally, headlines that mentioned popular celebrities being infected with a virus or getting injured, such as "(a Footballer player tested positive for coronavirus COVID-19 disease)", were often classified as health-related news, when the ground-truth label is sports. These examples highlight the complexity of the classification task and the need for robust models that can handle such ambiguities. Overall, our findings suggest that the proposed TransConvNet model is a promising approach for classifying short-text data, but further improvements may be necessary to achieve more accurate results in certain cases.

6.5 Conclusion

In this chapter, we present a comprehensive comparative study of effective deep learning models for Arabic text classification for both short and long documents in multi-class and multi-label scenarios. A total of 13 deep learning models were put into experimental evaluation on 6 large datasets to determine which model is best suitable for these tasks. We contribute to this matter, by proposing a new TransConvNet model that showed promising results across all experiments. Additionally, we present to the research community the largest, to our knowledge, Arabic text dataset for short-text domains.

Our experimental results reveal that fine-tuning BERT-based models, that was pretrained on

large Arabic text corpora outperform all neural-based model by a significant margin. Moreover, using long text sequences in model training can increase the complexity of the model and result in longer training times, in contrast to using short text data.

The complex nature of the Arabic language necessitates the use of deep and complex models to achieve high performance. This superior performance is likely due to the bidirectional structure of BERT models, contextual embeddings, and their tokenization technique to cover and represent unseen tokens while training. We hope that the release of our short-text dataset will help to advance the state of the art in text classification for Arabic data. Overall, our work contributes to the growing knowledge on Arabic text classification and provides a valuable resource for researchers in this field.

Conclusions

This PhD thesis addresses multiple research problems with different NLP applications, by creating new resources and developing novel effective deep learning models for Arabic language understanding and generation, using speech and text data in both MSA and dialectal Arabic. The research objectives were to contribute to the Arabic NLP research community by studying cutting-edge deep learning models and proposing methods that are tailored to Arabic language-specific characteristics.

Throughout the thesis, various NLP applications were studied, including sentiment analysis, hate speech detection in both Arabic speech and text corpora, Arabic speech emotion recognition, Arabic dialects identification, a new pretrained language model for the Algerian dialect, and Arabic news document classification. After analyzing the nature and complexity of the Arabic text and speech, it was necessary to use complex and specific models to achieve high performance across these applications. The Thesis is divided into five chapters, where each chapter discusses our contributions to a specific NLP application. In the first chapter, we present a nutshell of the SOTA deep learning models, as well as word representation techniques and current Arabic language models.

The first contribution of this thesis focuses on the analysis of text and speech in three distinct domains: Arabic sentiment analysis, hate speech detection, and speech emotion recognition. First, we introduce a novel neural network architecture for sentiment analysis and evaluate its performance on three different sentiment analysis datasets, where we demonstrate the superior performance of our methodology towards solving ASA compared to existing models. Secondly, we present our initial research efforts toward effective emotion recognition in Algerian dialect speech, in which we collect and annotate a new SER dataset from Algerian TV shows, and evaluated multiple classification models. Our SER methodology suggests that using a hybrid

LSTM-CNN model tends to achieve remarkable emotion detection performance and opens new application opportunities for the automation of emotion analysis through speech records in the Algerian dialect. Finally, we delve into the examination of potential risks and consequences associated with various forms of inappropriate speech, such as hate speech, abuse, bullying, and curse speech, across multiple dialects. Our investigation encompasses the identification and analysis of these speech categories in Modern Standard Arabic (MSA), Tunisian, Libyan, Saudi, and Khaliji dialects.

To address this issue, we propose a novel approach that combines a bidirectional LSTM-based model with a soft attention mechanism layer. Our methodology showcases remarkable performance improvements compared to previous works across multiple datasets, including those in MSA, Tunisian, Libyan, Saudi, and Khaliji dialects. Furthermore, recognizing the importance of covering the Algerian dialect, we present a comprehensive dataset collected from diverse online sources. This dataset is meticulously labeled to specifically address the detection of hate speech in the Algerian dialect, contributing to a more holistic understanding of the issue. Our proposed approaches for the aforementioned contributions yield remarkable performance achieving an accuracy of up to 96.29% in ASA, 93.34% in SER, and 97.47% in inappropriate speech detection.

The second contribution concentrates on the significance of Arabic dialect identification, and its potential uses for various commercial systems. We thoroughly examine the identification process for different Arabic dialects, using a carefully collected dataset tailored explicitly for the Maghrebi dialect. Additionally, we leverage existing datasets and employ diverse classification models, including classical machine learning models, deep learning architectures, and BERT-based models. Our findings demonstrate that BERT models outperform other classification techniques. This superiority can be attributed to the large pretraining corpus size and the effective word representation capabilities of the BERT models. Utilizing BERT enables more accurate and robust identification of Arabic dialects. The results obtained for Maghreb dialect identification tasks demonstrate a high identification accuracy of 97.93%.

In our third contribution, we highlight our methodology towards pretraining a new language model which we name DziriBERT that we pretrained on a large collected corpus of Algerian dialect composed of 3.3M sentences scraped from online social media platforms. The specific nature of the Algerian dialect and its various spoken forms on social media platforms using

Arabic and Arabizi text format pose significant challenges even for large-scale Arabic-BERT models.

To solve these challenges and advance the research applications in Algerian NLP, we pre-train DziriBERT using masked language modeling and adopted a BERT-like architecture on large cloud GPUs. The obtained results show that pre-training a dedicated model on a small dataset (450 MB) can outperform existing models that have been trained on much more data (hundreds of GB) across various NLP tasks in the Algerian dialect, including sentiment analysis, text classification, hate speech detection, and named entity recognition.

In our last and final contribution, we delve into the domain of Arabic news document classification from multiple perspectives. We address both long and short news document classification scenarios, considering both multi-class and multi-label text classification tasks. A significant portion of our research efforts focused on the collection of a substantial dataset specifically tailored for short texts in Arabic. Additionally, we introduce a novel transformer-based model called TransConvNet. Through extensive classification experiments, we evaluate the performance of various models, including neural-net models and BERT-based models. Our proposed TransConvNet model demonstrates superior performance compared to neural-net models across six Arabic classification datasets. However, it is worth noting that BERT models remain the top-performing models in this domain. Our research contributes to advancing Arabic news document classification by introducing a new model, exploring different classification scenarios, and providing insights into the performance of various models. These findings further our understanding of effective approaches for Arabic news document classification and pave the way for future advancements in this field.

Our experiments showed that contextual BERT embedding models, with their bidirectional structure, and tokenization technique, were the most effective models for Arabic language understanding and generation. Additionally, our curated datasets for hate speech detection, dialect identification, text classification, and language model pretraining have significantly contributed to the performance of the models

In summary, this PhD has contributed significantly to the field of Arabic NLP by proposing novel deep learning models, studying various NLP applications, and releasing new datasets. We believe that the findings of the research presented in this thesis will have practical implications for many applications, including social media analysis, document classification, and language

understanding. It is hoped that this work will inspire further research in this field and help researchers develop more efficient and effective NLP models for Arabic language understanding and generation.

References

- A.Meftah, Y.Alotaibi and S.Selouani (Nov. 2014). “Designing, Building, and Analyzing an Arabic Speech Emotional Corpus: Phase 2”. In: *5th International Conference on Arabic Language Processing*, pp. 181–184.
- Abbes, Ines, Zaghouani, Wajdi, El-Hardlo, Omaima, and Ashour, Faten (May 2020). “DAICT: A Dialectal Arabic Irony Corpus Extracted from Twitter”. In: *Proceedings of the 12th Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, pp. 6265–6271. URL: <https://aclanthology.org/2020.lrec-1.768>.
- Abdaoui, Amine, Berrimi, Mohamed, Oussalah, Mourad, and Moussaoui, Abdelouahab (2021). *DziriBERT: a Pre-trained Language Model for the Algerian Dialect*. DOI: [10.48550/ARXIV.2109.12346](https://arxiv.org/abs/2109.12346). URL: <https://arxiv.org/abs/2109.12346>.
- Abdaoui, Amine, Pradel, Camille, and Sigel, Grégoire (2020). “Load What You Need: Smaller Versions of Multilingual BERT”. In: *Proceedings of SustainNLP: Workshop on Simple and Efficient Natural Language Processing @EMNLP*, pp. 119–123. DOI: [10.18653/v1/2020.sustainlp-1.16](https://www.aclweb.org/anthology/2020.sustainlp-1.16). URL: <https://www.aclweb.org/anthology/2020.sustainlp-1.16>.
- Abdelali, Ahmed, Hassan, Sabit, Mubarak, Hamdy, Darwish, Kareem, and Samih, Younes (2021). “Pre-Training BERT on Arabic Tweets: Practical Considerations”. In: *ArXiv preprint arXiv:2102.10684* abs/2102.10684. URL: <https://arxiv.org/abs/2102.10684>.
- Abdelali, Ahmed, Mubarak, Hamdy, Samih, Younes, Hassan, Sabit, and Darwish, Kareem (June 2021). “QADI: Arabic Dialect Identification in the Wild”. In: *Proceedings of the Sixth Arabic Natural Language Processing Workshop*. Kyiv, Ukraine (Virtual): Association for Computational Linguistics, pp. 1–10. URL: <https://aclanthology.org/2021.wanlp-1.1>.

- Abdul-Mageed, Muhammad, Elmadany, AbdelRahim, and Nagoudi, El Moatez Billah (Aug. 2021). “ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, pp. 7088–7105. DOI: [10.18653/v1/2021.acl-long.551](https://doi.org/10.18653/v1/2021.acl-long.551). URL: <https://aclanthology.org/2021.acl-long.551>.
- Abdul-Mageed, Muhammad, Elmadany, AbdelRahim A., and Nagoudi, El Moatez Billah (2021). “ARBERT & MARBERT: Deep Bidirectional Transformers for Arabic”. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pp. 7088–7105. URL: <https://aclanthology.org/2021.acl-long.551.pdf>.
- Abdul-Mageed, Muhammad, Zhang, Chiyu, Bouamor, Houda, and Habash, Nizar (Dec. 2020). “NADI 2020: The First Nuanced Arabic Dialect Identification Shared Task”. In: *Proceedings of the Fifth Arabic Natural Language Processing Workshop*. Barcelona, Spain (Online): Association for Computational Linguistics, pp. 97–110. URL: <https://aclanthology.org/2020.wanlp-1.9>.
- Abdul-Mageed, Muhammad, Zhang, Chiyu, Elmadany, AbdelRahim, Bouamor, Houda, and Habash, Nizar (June 2021). “NADI 2021: The Second Nuanced Arabic Dialect Identification Shared Task”. In: *Proceedings of the Sixth Arabic Natural Language Processing Workshop*. Kyiv, Ukraine (Virtual): Association for Computational Linguistics, pp. 244–259. URL: <https://aclanthology.org/2021.wanlp-1.28>.
- (Dec. 2022). “NADI 2022: The Third Nuanced Arabic Dialect Identification Shared Task”. In: *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*. Ed. by Houda Bouamor, Hend Al-Khalifa, Kareem Darwish, Owen Rambow, Fethi Bougares, Ahmed Abdelali, Nadi Tomeh, Salam Khalifa, and Wajdi Zaghouni. Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, pp. 85–97. DOI: [10.18653/v1/2022.wanlp-1.9](https://doi.org/10.18653/v1/2022.wanlp-1.9). URL: <https://aclanthology.org/2022.wanlp-1.9>.
- Abdullah, I, Alharbi, Phillip Smith, and Mark, Lee (2021). “Enhancing contextualised language models with static character and word embeddings for emotional intensity and sentiment strength detection in arabic tweets”. In: *Procedia, Computer Science* 189, pp. 258–265.

- Abu Farha, Ibrahim and Magdy, Walid (2019). “Mazajak: An Online Arabic Sentiment Analyser”. In: *Proceedings of the Fourth Arabic Natural Language Processing Workshop*. Association for Computational Linguistics, pp. 192–198. DOI: [10.18653/v1/W19-4621](https://doi.org/10.18653/v1/W19-4621).
- Abu Kwaik, Kathrein, Saad, Motaz K, Chatzikyriakidis, Stergios, and Dobnik, Simon (2019). “LSTM-CNN Deep Learning Model for Sentiment Analysis of Dialectal Arabic”. In: *Proceedings of the International Conference on Arabic Language Processing*. Vol. 1108. Springer Science and Business Media LLC, pp. 108–121.
- Abualigah, Laith (Dec. 2018). *Feature Selection and Enhanced Krill Herd Algorithm for Text Document Clustering*. DOI: [10.1007/978-3-030-10674-4](https://doi.org/10.1007/978-3-030-10674-4).
- Abualigah, Laith, Khader, Ahamad Tajudin, Al-Betar, Mohammed, and Awadallah, Mohammed (May 2016). “A Krill Herd Algorithm For Efficient Text Documents Clustering”. In: DOI: [10.1109/ISCAIE.2016.7575039](https://doi.org/10.1109/ISCAIE.2016.7575039).
- Al Sallab, Ahmad, Hajj, Hazem, Badaro, Gilbert, Baly, Ramy, El-Hajj, Wassim, and Shaban, Khaled (2015). “Deep learning models for sentiment analysis in Arabic”. In: *Proceedings of the second workshop on Arabic natural language processing*, pp. 9–17.
- Al Sbou, Ahed MF (2018). “A survey of arabic text classification models”. In: *International Journal of Electrical and Computer Engineering (IJECE)* 8.6, pp. 4352–4355.
- El-Alami, Fatima-zahra, El Alaoui, Said Ouatik, and Nahnahi, Nouredine En (2021). “Contextual semantic embeddings based on fine-tuned AraBERT model for Arabic text multi-class categorization”. In: *Journal of King Saud University-Computer and Information Sciences*.
- Alammary, Ali Saleh (2022). “BERT Models for Arabic Text Classification: A Systematic Review”. In: *Applied Sciences* 12.11, p. 5720.
- Alamro, Hind, Alshehri, Manal, Alharbi, Basma, Khayyat, Zuhair, Kalkatawi, Manal, Jaber, Inji Ibrahim, and Zhang, Xiangliang (2021). “Overview of the Arabic Sentiment Analysis 2021 Competition at KAUST”. In: *CoRR* abs/2109.14456. arXiv: [2109.14456](https://arxiv.org/abs/2109.14456), URL: <https://arxiv.org/abs/2109.14456>.

- Alayba, Abdulaziz M, Palade, Vasile, England, Matthew, and Iqbal, Rahat (2018). “A combined CNN and LSTM model for arabic sentiment analysis”. In: *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*. Springer, pp. 179–191.
- Albadi, N., Kurdi, M., and Mishra, S. (2018). “Are they our brothers? Analysis and detection of religious hate speech in the Arabic Twittersphere”. In: *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, pp. 69–76.
- Alceu S. Britto Robert Sabourinc, Luiz E.S. Oliveira (2014). “Dynamic selection of classifiers—A comprehensive review”. In: *Pattern Recognition* 47, pp. 3665–3680. DOI: <https://doi.org/10.1016/j.patcog.2014.05.003>.
- Ali, Raza, Farooq, Umar, Arshad, Umair, Shahzad, Waseem, and Beg, Mirza Omer (2022). “Hate speech detection on Twitter using transfer learning”. In: *Computer Speech & Language* 74, p. 101365. ISSN: 0885-2308. DOI: <https://doi.org/10.1016/j.csl.2022.101365>. URL: <https://www.sciencedirect.com/science/article/pii/S0885230822000110>.
- Almani, Nada and Tang, Lillian (Mar. 2020). “Deep Attention-Based Review Level Sentiment Analysis for Arabic Reviews”. In: pp. 47–53. DOI: [10.1109/CDMA47397.2020.00014](https://doi.org/10.1109/CDMA47397.2020.00014).
- Alotaiby, Fahad (2011). “Automatic headline generation using character cross-correlation”. In: *Proceedings of the ACL 2011 Student Session*, pp. 117–121.
- Alshaalan, Raghad and Al-Khalifa, Hend (2020). “Hate Speech Detection in Saudi Twittersphere: A Deep Learning Approach”. In: *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pp. 12–23.
- Alshehri, Ali, Nagoudi, El Moatez Billah, and Abdul-Mageed, Muhammad (May 2020). “Understanding and Detecting Dangerous Speech in Social Media”. In: *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*. Marseille, France: European Language Resource Association, pp. 40–47. ISBN: 979-10-95546-51-1. URL: <https://aclanthology.org/2020.osact-1.6>.
- Alsmadi, Issa and Gan, Keng Hoon (2019). “Review of short-text classification”. In: *International Journal of Web Information Systems*.

- Altowayan, A Aziz and Tao, Lixin (2016). “Word embeddings for Arabic sentiment analysis”. In: *IEEE International Conference on Big Data (Big Data)*. IEEE, pp. 3820–3825.
- Alwaneen, Tahani H, Azmi, Aqil M, Aboalsamh, Hatim A, Cambria, Erik, and Hussain, Amir (2022). “Arabic question answering system: a survey”. In: *Artificial Intelligence Review* 55.1, pp. 207–253.
- Aly, Mohamed and Atiya, Amir (2013). “Labr: A large scale arabic book reviews dataset”. In: *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 494–498.
- Alzawaydeh, Rashid and Alghazo, Sharif (Dec. 2018). “Analysing Media Discourse: The Case of Conceptual Metaphors in Football News Headlines in English and Arabic”. In: 10, p. 116. DOI: [10.5296/ijl.v10i6.13502](https://doi.org/10.5296/ijl.v10i6.13502).
- Antoun, Wissam, Baly, Fady, and Hajj, Hazem (May 2020). “AraBERT: Transformer-based Model for Arabic Language Understanding”. In: *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*. Marseille, France: European Language Resource Association, pp. 9–15. URL: <https://aclanthology.org/2020.osact-1.2>.
- Ayadi, M. El, Kamel, M., and Karray, F. (Apr. 2017). “Speech emotion recognition using Gaussian mixture vector autoregressive models”. In: *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*. Vol. 4, pp. IV–957.
- Al-Azani, Sadam and El-Alfy, El-Sayed (2018). “Emojis-based sentiment classification of Arabic microblogs using deep recurrent neural networks”. In: *2018 International Conference on Computing Sciences and Engineering (ICCSE)*. IEEE, pp. 1–6.
- Bahdanau, Dzmitry, Cho, Kyunghyun, and Bengio, Yoshua (2014a). *Neural Machine Translation by Jointly Learning to Align and Translate*. DOI: [10.48550/ARXIV.1409.0473](https://doi.org/10.48550/ARXIV.1409.0473). URL: <https://arxiv.org/abs/1409.0473>.
- (2014b). “Neural machine translation by jointly learning to align and translate”. In: *arXiv preprint arXiv:1409.0473*.

- Basma, Alharbi, Hind, Alamro, Manal, Alshehri, Zuhair, Khayyat, Manal, Kalkatawi, Inji, Ibrahim Jaber, and Xiangliang, Zhang (2021). *ASAD: A Twitter-based Benchmark Arabic Sentiment Analysis Dataset*. arXiv: [2011.00578 \[cs.CL\]](https://arxiv.org/abs/2011.00578).
- El-Beltagy, Samhaa R., El Kalamawy, Mona, and Soliman, Abu Bakr (2017). “NileTMRG at SemEval-2017 Task 4: Arabic Sentiment Analysis”. In: *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. Vancouver, Canada: Association for Computational Linguistics, pp. 790–795. DOI: [10.18653/v1/S17-2133](https://doi.org/10.18653/v1/S17-2133).
- Bengio, Yoshua, Simard, Patrice, and Frasconi, Paolo (1994). “Learning long-term dependencies with gradient descent is difficult”. In: *IEEE Transactions on Neural Networks* 5.2, pp. 157–166.
- Berrimi, Mohamed, Moussaoui, Abdelouahab, Oussalah, Mourad, and Saidi, Mohamed (2020). “Arabic dialects identification: North African dialects case study”. In.
- Berrimi, Mohamed and Moussaoui, Abdelouaheb (2020). “Deep learning for identifying and classifying retinal diseases”. In: *2020 2nd International Conference on Computer and Information Sciences (ICCIS)*, pp. 1–6.
- Berrimi, Mohamed, Moussaoui, Abdelouaheb, Oussalah, Mourad, and Saidi, Mohamed (2020a). “Arabic dialects identification: North African dialects case study”. In: *THIRD CONFERENCE ON INFORMATICS AND APPLIED MATHEMATICS IAM’20*.
- (2020b). “Attention-based networks for analyzing inappropriate speech in Arabic text”. In: *2020 4th International Symposium on Informatics and its Applications (ISIA)*, pp. 1–6. DOI: [10.1109/ISIA51297.2020.9416539](https://doi.org/10.1109/ISIA51297.2020.9416539).
- Berrimi, Mohamed, Oussalah, Mourad, Moussaoui, Abdelouahab, and Saidi, Mohamed (2022). “Attention Mechanism Architecture for Arabic Sentiment Analysis”. In: *ACM Transactions on Asian and Low-Resource Language Information Processing*.
- (2023). “A Comparative Study of Effective Approaches for Arabic Text Classification”. In: *Available at SSRN 4361591*.

- Bottino, S. M. B., Bottino, C. M. C., Regina, C. G., Correia, A. V. L., and Ribeiro, W. S. (2015). “Cyberbullying and adolescent mental health: systematic review”. In: *Cadernos de Sa’ude P’ublica* 31.3, pp. 463–475. DOI: [10.1590/0102-311x00036114](https://doi.org/10.1590/0102-311x00036114).
- Bouamor, Houda, Habash, Nizar, Salameh, Mohammad, Zaghoulani, Wajdi, Rambow, Owen, Abdulrahim, Dana, Obeid, Ossama, Khalifa, Salam, Eryani, Fadhl, Erdmann, Alexander, et al. (2018). “The MADAR Arabic Dialect Corpus and Lexicon.” In: *LREC*.
- Bouamor, Houda, Hassan, Sabit, and Habash, Nizar (2019). “The MADAR Shared Task on Arabic Fine-Grained Dialect Identification”. In: *Proceedings of the Fourth Arabic Natural Language Processing Workshop*. Florence, Italy: Association for Computational Linguistics, pp. 199–207. DOI: [10.18653/v1/W19-4622](https://doi.org/10.18653/v1/W19-4622). URL: <https://aclanthology.org/W19-4622>.
- Boudad, Naaima, Faizi, Rdouan, Thami, Rachid Oulad Haj, and Chiheb, Raddouane (2018). “Sentiment analysis in Arabic: A review of the literature”. In: *Ain Shams Engineering Journal* 9.4, pp. 2479–2490.
- Boukil, Samir, Biniz, Mohamed, El Adnani, Fatiha, Cherrat, Loubna, and El Moutaouakkil, Abd Elmajid (2018). “Arabic text classification using deep learning technics”. In: *International Journal of Grid and Distributed Computing* 11.9, pp. 103–114.
- Brandesen, Alex, Verberne, Suzan, Lambers, Karsten, and Wansleeben, Milco (2022). “Can BERT Dig It?—named entity recognition for information retrieval in the archaeology domain”. In: *Journal on Computing and Cultural Heritage (JOCCH)*.
- Brown, Tom, Mann, Benjamin, Ryder, Nick, Subbiah, Melanie, Kaplan, Jared D, Dhariwal, Prafulla, Neelakantan, Arvind, Shyam, Pranav, Sastry, Girish, Askell, Amanda, Agarwal, Sandhini, Herbert-Voss, Ariel, Krueger, Gretchen, Henighan, Tom, Child, Rewon, Ramesh, Aditya, Ziegler, Daniel, Wu, Jeffrey, Winter, Clemens, Hesse, Chris, Chen, Mark, Sigler, Eric, Litwin, Mateusz, Gray, Scott, Chess, Benjamin, Clark, Jack, Berner, Christopher, McCandlish, Sam, Radford, Alec, Sutskever, Ilya, and Amodei, Dario (2020). “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin. Vol. 33, pp. 1877–1901. URL: <https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfc4967418bfb8ac142f64a-Paper.pdf>.

- Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., and Weiss, B. (2005). “A database of German emotional speech”. In: *Ninth European Conference on Speech Communication and Technology*.
- Busso, Carlos, Bulut, Murtaza, Lee, Chi-Chun, Kazemzadeh, Abe, Mower, Emily, Kim, Samuel, Chang, Jeannette N, Lee, Sungbok, and Narayanan, Shrikanth S (2008). “IEMOCAP: Interactive emotional dyadic motion capture database”. In: *Language resources and evaluation* 42, pp. 335–359.
- C.Li Z.Bao, L.Li and Z.Zhao (May 2020). “Exploring Temporal Representations by Leveraging Attention-based Bidirectional LSTM-RNNs for Multi-modal Emotion Recognition”. In: *Information Processing & Management* 57.3, p. 102185. DOI: [10.1016/j.ipm.2019.102185](https://doi.org/10.1016/j.ipm.2019.102185).
- Chatterjee, Rajdeep, Mazumdar, Saptarshi, Sherratt, R Simon, Halder, Rohit, Maitra, Tanmoy, and Giri, Debasis (2021). “Real-time speech emotion analysis for smart home assistants”. In: *IEEE Transactions on Consumer Electronics* 67.1, pp. 68–76.
- Chatzakou, D., Leontiadis, I., Blackburn, J., Cristofaro, E. D., Stringhini, G., Vakali, A., and Kourtellis, N. (2019). “Detecting cyberbullying and cyberaggression in social media”. In: *ACM Transactions on the Web (TWEB)* 13.3, pp. 1–51.
- Chen, Mengen, Jin, Xiaoming, and Shen, Dou (2011). “Short text classification improved by learning multi-granularity topics”. In: *Twenty-second international joint conference on artificial intelligence*. Citeseer.
- Cheng, X. and Duan, Q. (2012). “Speech emotion recognition using gaussian mixture model”. In: *Proceedings of the 2012 International Conference on Computer Application and System Modeling*, pp. 1222–1225.
- Chouikhi, Hasna, Chniter, Hamza, and Jarray, Fethi (2021). “Arabic sentiment analysis using BERT model”. In: *International Conference on Computational Collective Intelligence*. Springer, pp. 621–632.
- Chowdhury, A. G., Didolkar, A., Sawhney, R., and Shah, R. (July 2019). “ARHNet-Leveraging Community Interaction for Detection of Religious Hate Speech in Arabic”. In: *Proceedings of*

the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pp. 273–280.

Chowdhury, Shammur Absar, Abdelali, Ahmed, Darwish, Kareem, Soon-Gyo, Jung, Salminen, Joni, and Jansen, Bernard J (2020). “Improving Arabic text categorization using transformer training diversification”. In: *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pp. 226–236.

Chung, Junyoung, Gulcehre, Caglar, Cho, KyungHyun, and Bengio, Yoshua (2014). “Empirical evaluation of gated recurrent neural networks on sequence modeling”. In: *arXiv preprint arXiv:1412.3555*.

Conneau, Alexis, Khandelwal, Kartikay, Goyal, Naman, Chaudhary, Vishrav, Wenzek, Guillaume, Guzmán, Francisco, Grave, Édouard, Ott, Myle, Zettlemoyer, Luke, and Stoyanov, Veselin (2020). “Unsupervised Cross-lingual Representation Learning at Scale”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 8440–8451. URL: <https://www.aclweb.org/anthology/2020.acl-main.747.pdf>.

Correa, J. A. Miranda, Abadi, M. K., Sebe, N., and Patras, I. (2018). “AMIGOS: A Dataset for Affect, Personality and Mood Research on Individuals and Groups”. In: *IEEE Transactions on Affective Computing*, pp. 1–1. DOI: [10.1109/taffc.2018.2884461](https://doi.org/10.1109/taffc.2018.2884461).

Cotterell, Ryan and Callison-Burch, Chris (May 2014). “A Multi-Dialect, Multi-Genre Corpus of Informal Written Arabic”. In: *LREC*, pp. 241–245.

Al-Dabet, Saja and Tedmori, Sara (2019). “Sentiment Analysis for Arabic Language using Attention-Based Simple Recurrent Unit”. In: *2nd International Conference on new Trends in Computing Sciences (ICTCS)*. IEEE, pp. 1–6.

Al-Dabet, Saja, Tedmori, Sara, and AL-Smadi, Mohammad (2021). “Enhancing Arabic aspect-based sentiment analysis using deep learning models”. In: *Computer Speech & Language* 69, p. 101224. ISSN: 0885-2308. DOI: <https://doi.org/10.1016/j.csl.2021.101224>. URL: <https://www.sciencedirect.com/science/article/pii/S0885230821000310>.

- Dahmani, H., Hussein, H., Meyer-Sickendiek, B., and Jokisch, O. (Oct. 2019). “Natural Arabic Language Resources for Emotion Recognition in Algerian Dialect”. In: *International Conference on Arabic Language Processing*, pp. 18–33.
- Dahou, Abdelghani, Xiong, Shengwu, Zhou, Junwei, and Elaziz, M. A. (2019). “Multi-Channel Embedding Convolutional Neural Network Model for Arabic Sentiment Classification”. In: *ACM Transactions on Asian Low Resources and Language Information Processing* 18, pp. 1–41.
- Dahou, Abdelhalim Hafedh and Cheragui, Mohamed Amine (2023a). “Impact of Normalization and Data Augmentation in NER for Algerian Arabic Dialect”. In: *Modelling and Implementation of Complex Systems*. Ed. by Salim Chikhi, Gregorio Diaz-Descalzo, Abdelmalek Amine, Allaoua Chaoui, Djamel Eddine Saidouni, and Mohamed Khireddine Kholadi. Cham: Springer International Publishing, pp. 249–262. ISBN: 978-3-031-18516-8.
- (2023b). “Named Entity Recognition for Algerian Arabic Dialect in Social Media”. In: *12th International Conference on Information Systems and Advanced Technologies “ICISAT 2022”*. Ed. by Mohamed Ridha Laouar, Valentina Emilia Balas, Brahim Lejdel, Sean Eom, and Mohamed Amine Boudia. Cham: Springer International Publishing, pp. 135–145. ISBN: 978-3-031-25344-7.
- Darwish, Kareem, Habash, Nizar, Abbas, Mourad, Al-Khalifa, Hend, Al-Natsheh, Husein T, El-Beltagy, Samhaa R, Bouamor, Houda, Bouzoubaa, Karim, Cavalli-Sforza, Violetta, El-Hajj, Wassim, et al. (2020). “A Panoramic Survey of Natural Language Processing in the Arab World”. In: *arXiv preprint arXiv:2011.12631*.
- Darwish, Khaled (2013). “Arabizi detection and conversion to Arabic”. In: *arXiv preprint arXiv:1306.6755*.
- Deng, Jianfeng, Cheng, Lianglun, and Wang, Zhuowei (2021). “Attention-based BiLSTM fused CNN with gating mechanism model for Chinese long text classification”. In: *Computer Speech & Language* 68, p. 101182. ISSN: 0885-2308. DOI: <https://doi.org/10.1016/j.csl.2020.101182>. URL: <https://www.sciencedirect.com/science/article/pii/S0885230820301157>.

- Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton, and Toutanova, Kristina (2018). “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805*.
- (June 2019). “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186. DOI: [10.18653/v1/N19-1423](https://doi.org/10.18653/v1/N19-1423). URL: <https://www.aclweb.org/anthology/N19-1423>.
- Egger, Maria, Ley, Matthias, and Hanke, Sten (2019). “Emotion recognition from physiological signal analysis: A review”. In: *Electronic Notes in Theoretical Computer Science* 343, pp. 35–55.
- Einea, Omar, Elnagar, Ashraf, and Al Debsi, Ridhwan (2019). “Sanad: Single-label arabic news articles dataset for automatic text categorization”. In: *Data in brief* 25, p. 104076.
- El Rifai, Hozayfa, Al Qadi, Leen, and Elnagar, Ashraf (Mar. 2021). “Arabic Multi-label Text Classification of News Articles”. In: pp. 431–444. ISBN: 978-3-030-69716-7. DOI: [10.1007/978-3-030-69717-4_41](https://doi.org/10.1007/978-3-030-69717-4_41).
- Eljawad, L., Aljamaeen, R., Alsmadi, M., Almarashdeh, I., Abouelmagd, H., Alsmadi, S., Haddad, F., Alkhasawneh, R. A., Alzughoul, M., and Alazzam, M. B. (2019). “Arabic Voice Recognition Using Fuzzy Logic and Neural Network”. In: *ELJAWAD, L., ALJAMAEEN, R., ALSMADI, MK, AL-MARASHDEH, I., ABOUELMAGD, H., ALSMADI, S., HADDAD, F., ALKHASAWNEH, RA, ALZUGHOUL, M. & ALAZZAM, MB*, pp. 651–662.
- ElJundi, Obeida, Antoun, Wissam, El Droubi, Nour, Hajj, Hazem, El-Hajj, Wassim, and Shaban, Khaled (Aug. 2019). “hULMonA: The Universal Language Model in Arabic”. In: *Proceedings of the Fourth Arabic Natural Language Processing Workshop*. Florence, Italy: Association for Computational Linguistics, pp. 68–77. DOI: [10.18653/v1/W19-4608](https://doi.org/10.18653/v1/W19-4608). URL: <https://www.aclweb.org/anthology/W19-4608>.
- Elmadany, A., Zhang, C., Abdul-Mageed, M., and Hashemi, A. (2020). “Leveraging Affective Bidirectional Transformers for Offensive Language Detection”. In: eprint: [arXiv:2006.01266](https://arxiv.org/abs/2006.01266).

- Elmadany, AbdelRahim, Mubarak, Hamdy, and Magdy, Walid (2018). “Arsas: An arabic speech-act and sentiment corpus of tweets”. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC)*, pp. 1–6.
- Elmadany, AbdelRahim, Nagoudi, El Moatez Billah, and Abdul-Mageed, Muhammad (2022). “ORCA: A Challenging Benchmark for Arabic Language Understanding”. In: *arXiv preprint arXiv:2212.10758*.
- Elnagar, Ashraf and Einea, Omar (2016). “BRAD 1.0: Book reviews in Arabic dataset”. In: *IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, pp. 1–8.
- Elnagar, Ashraf, Khalifa, Yasmin, and Einea, Anas (2018). “Hotel Arabic-Reviews Dataset Construction for Sentiment Analysis Applications”. In: pp. 35–52. ISBN: 978-3-319-67055-3. DOI: [10.1007/978-3-319-67056-0_3](https://doi.org/10.1007/978-3-319-67056-0_3).
- ElSahar, Hady and El-Beltagy, S. (2015). “Building Large Arabic Multi-domain Resources for Sentiment Analysis”. In: *Proceedings of the International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*, pp. 23–34.
- ElSahar, Hady and El-Beltagy, Samhaa (2015). “Building Large Arabic Multi-domain Resources for Sentiment Analysis”. In: *Lecture Notes in Computer Science* 9042, pp. 23–34. DOI: [10.1007/978-3-319-18117-2_2](https://doi.org/10.1007/978-3-319-18117-2_2).
- Elsayed, Hoda Ahmed Galal, Chaffar, Soumaya, Belhaouari, Samir Brahim, and Raissouli, Hafsa (2020). “A two-level deep learning approach for emotion recognition in Arabic news headlines”. In: *International Journal of Computers and Applications* 0.0, pp. 1–10. DOI: [10.1080/1206212X.2020.1851501](https://doi.org/10.1080/1206212X.2020.1851501), eprint: <https://doi.org/10.1080/1206212X.2020.1851501>, URL: <https://doi.org/10.1080/1206212X.2020.1851501>.
- Al-Faham, A. and Ghneim, N. (2016). “Towards enhanced arabic speech emotion recognition: comparison between three methodologies”. In: *Asian J. Sci. Technol* 7.3, pp. 2665–2669.
- Farha, Ibrahim Abu and Magdy, Walid (2021a). “A comparative study of effective approaches for Arabic sentiment analysis”. In: *Information Processing & Management* 58.2, p. 102438.

- (2021b). “A comparative study of effective approaches for Arabic sentiment analysis”. In: *Information Processing & Management* 58.2, p. 102438.
- Fayek, Haytham M, Lech, Margaret, and Cavedon, Lawrence (2017). “Evaluating deep learning architectures for speech emotion recognition”. In: *Neural Networks* 92, pp. 60–68.
- Feldman, R. and Sanger, J. (2006). *The Text Mining Handbook*. Cambridge University Press.
- Gamal, Donia, Alfonse, Marco, El-Horbaty, El-Sayed M., and Salem, Abdel-Badeeh M. (2019). “Implementation of Machine Learning Algorithms in Arabic Sentiment Analysis Using N-Gram Features”. In: *Procedia Computer Science, ICICT* 154, pp. 332–340. DOI: <https://doi.org/10.1016/j.procs.2019.06.048>.
- Goodfellow, Ian, Bengio, Yoshua, Courville, Aaron, and Bengio, Yoshua (2016). *Deep learning*. Vol. 1. 2. MIT press Cambridge.
- Guellil, I. and Azouaou, F. (2017). “Arabic dialect identification with an unsupervised learning (based on a lexicon). application case: Algerian dialect”. In: *IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC) and 15th International Symposium on Distributed Computing and Applications for Business Engineering (DCABES)*, pp. 724–731.
- Guellil, Imane, Adeel, Ahsan, Azouaou, Faical, Boubred, Mohamed, Houichi, Yousra, and Moumna, Akram Abdelhaq (2021). *Seixism detection: The first corpus in Algerian dialect with a code-switching in Arabic/ French and English*. DOI: [10.48550/ARXIV.2104.01443](https://arxiv.org/abs/2104.01443). URL: <https://arxiv.org/abs/2104.01443>.
- Guellil, Imane, Saadane, Houda, Azouaou, Faical, Gueni, Billel, and Nouvel, Damien (2021). “Arabic Natural Language Processing: an overview”. In: *Journal of King Saud University - Computer and Information Sciences* 33(2), pp. 497–505. DOI: [10.1016/j.jksuci.2019.02.006](https://doi.org/10.1016/j.jksuci.2019.02.006).
- Guellil, Imène and Azouaou, Faiçal (2016). “Arabic Dialect Identification with an Unsupervised Learning (Based on a Lexicon). Application Case: ALGERIAN Dialect”. In: *2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed*

- Computing and Applications for Business Engineering (DCABES)*, pp. 724–731. DOI: [10.1109/CSE-EUC-DCABES.2016.268](https://doi.org/10.1109/CSE-EUC-DCABES.2016.268).
- Habash, Nizar (Jan. 2007). “Arabic Morphological Representations for Machine Translation”. In: vol. 38, pp. 263–285. ISBN: 978-1-4020-6045-8. DOI: [10.1007/978-1-4020-6046-5_14](https://doi.org/10.1007/978-1-4020-6046-5_14).
- (Apr. 2019). *Grammarly, Arabic Natural Language Processing: Challenges and Solutions*. URL: <https://grammarly.ai/arabic-natural-language-processing-challenges-and-solutions/>.
- Habash, Nizar Y (2010). “Introduction to Arabic natural language processing”. In: *Synthesis Lectures on Human Language Technologies 3.1*, pp. 1–187.
- Haddad, B., Orabe, Z., Al-Abood, A., and Ghneim, N. (2020). “Arabic Offensive Language Detection with Attention-based Deep Neural Networks”. In: *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pp. 76–81.
- Haddad, H., Mulki, H., and Oueslati, A. (2019). “THSAB: A Tunisian Hate Speech and Abusive Dataset”. In: *International Conference on Arabic Language Processing*. Springer, Cham, pp. 251–263.
- Haidar, Batoul, Maroun, Chamoun, and Serhrouchni, Ahmed (2018). “Arabic Cyberbullying Detection: Using Deep Learning”. In: *2018 IEEE Conference on Computer and Communications Engineering (ICCCE)*, pp. 284–289. DOI: [10.1109/ICCCE.2018.8539303](https://doi.org/10.1109/ICCCE.2018.8539303).
- Haider, Ahmad and Hussein, Riyad (Dec. 2019). “Analysing headlines as a way of downsizing news corpora: Evidence from an Arabic-English comparable corpus of newspaper articles”. In.
- Hammoud, Jaafar, Vatian, Aleksandra, Dobrenko, Natalia, Vedernikov, Nikolai, Shalyto, Anatoly, and Gusarova, Natalia (2021). “New Arabic Medical Dataset for Diseases Classification”. In: *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, pp. 196–203.

- Harar, P., Burget, R., and Dutta, M. (2017). “Speech emotion recognition with deep learning”. In: *2017 4th International Conference on Signal Processing and Integrated Networks (SPIN)*. Ed. by P. Harar, R. Burget, and M. Dutta, pp. 137–140.
- Harrat, Salima, Meftouh, Karima, and Smaili, Kamel (2019). “Machine translation for Arabic dialects (survey)”. In: *Information Processing & Management* 56.2, pp. 262–273.
- Heikal Maha Torki Marwan, Nagwa El-Makky (2018). “Sentiment Analysis of Arabic Tweets using Deep Learning”. In: *Procedia Computer Science*, 142 (2018), 114-122 142, pp. 114–122. DOI: <https://doi.org/10.1016/j.procs.2018.10.466>.
- Hochreiter, Sepp and Schmidhuber, Jürgen (1996). “LSTM can solve hard long time lag problems”. In: *Advances in neural information processing systems* 9.
- Hossain, MD Zakir, Sohel, Ferdous, Shiratuddin, Mohd Fairuz, and Laga, Hamid (2019). “A comprehensive survey of deep learning for image captioning”. In: *ACM Computing Surveys* 51.6, pp. 1–36.
- Howard, Jeremy and Ruder, Sebastian (2018). “Universal language model fine-tuning for text classification”. In: *arXiv preprint arXiv:1801.06146*.
- Hsu, Wei-Ning, Bolte, Benjamin, Tsai, Yao-Hung Hubert, Lakhota, Kushal, Salakhutdinov, Ruslan, and Mohamed, Abdelrahman (2021). “Hubert: Self-supervised speech representation learning by masked prediction of hidden units”. In: *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29, pp. 3451–3460.
- Hu, H., Xu, M. X., and Wu, W. (2007). “GMM supervector based SVM with spectral features for speech emotion recognition”. In: *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*. Vol. 4. IEEE, pp. IV–413.
- Huang, C. W. and Narayanan, S. S. (2017). “Deep convolutional recurrent neural network with attention mechanism for robust speech emotion recognition”. In: *2017 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, pp. 583–588.
- Huang, Lun, Wang, Wenmin, Chen, Jie, and Wei, Xiao-Yong (2019). “Attention on Attention for Image Captioning”. In.

- Ibrahim Abu Farha, Wajdi Zaghouni and Walid, Magdy (2021). “Overview of the wanlp 2021 shared task on sarcasm and sentiment detection in Arabic”. In: *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pp. 296–305.
- Inoue, Go, Alhafni, Bashar, Baimukan, Nurpeiis, Bouamor, Houda, and Habash, Nizar (June 2021). “The Interplay of Variant, Size, and Task Type in Arabic Pre-trained Language Models”. In: *Proceedings of the Sixth Arabic Natural Language Processing Workshop*. Kyiv, Ukraine (Virtual): Association for Computational Linguistics, pp. 92–104. URL: <https://aclanthology.org/2021.wanlp-1.10>.
- Ioffe, Sergey and Szegedy, Christian (2015). “Batch normalization: Accelerating deep network training by reducing internal covariate shift”. In: *International conference on machine learning*. pmlr, pp. 448–456.
- Jha, Nagesh, Jethva, Aakash, Parmar, Nidhi, and Patil, Abhay (2016). “A Review Paper on Deep Web Data Extraction using WordNet”. In: *International Research Journal of Engineering and Technology* 03(3), pp. 1003–1006.
- Kazarian, J. and Ammar, J. (2013). “School Bullying in the Arab World: A Review”. In: *The Arab Journal of Psychiatry* 24.1, pp. 37–45.
- Al-Khalifa, Hend, Magdy, Walid, Darwish, Kareem, Elsayed, Tamer, and Mubarak, Hamdy, eds. (May 2020). *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*. Marseille, France: European Language Resource Association. ISBN: 979-10-95546-51-1. URL: <https://aclanthology.org/2020.osact-1.0>.
- Khalifa, Salam, Zalmout, Nasser, and Habash, Nizar (2020). “Morphological analysis and disambiguation for Gulf Arabic: The interplay between resources and methods”. In: *Proceedings of The 12th Language Resources and Evaluation Conference*, pp. 3895–3904.
- Khalil, A., Al-Khatib, W., El-Alfy, E., and Cheded, L. (Mar. 2018). “Anger detection in arabic speech dialogs”. In: *2018 International Conference on Computing Sciences and Engineering (ICCSE)*, pp. 1–6.

- Khalil, Ruhul Amin, Jones, Edward, Babar, Mohammad Inayatullah, Jan, Tariqullah, Zafar, Mohammad Haseeb, and Alhussain, Thamer (2019). “Speech emotion recognition using deep learning techniques: A review”. In: *IEEE Access* 7, pp. 117327–117345.
- Khoja, Yehia, Alhadlaq, Omar, and Al-Saif, Saud (2017). “Auto Generation of Arabic News Headlines”. In.
- Kim, J. and Saurous, R. (Sept. 2018). “Emotion recognition from human speech using temporal information and deep learning”. In: *Interspeech*, pp. 937–940.
- Kim, Yoon (2014). *Convolutional Neural Networks for Sentence Classification*. arXiv: [1408.5882](https://arxiv.org/abs/1408.5882) [cs.CL].
- Klaylat, S., Osman, Z., Hamandi, L., and Zantout, R. (Mar. 2018). “Emotion recognition in Arabic speech”. In: *Analog Integrated Circuits and Signal Processing* 96.2, pp. 337–351. DOI: [10.1007/s10470-018-1142-4](https://doi.org/10.1007/s10470-018-1142-4).
- Kuncheva, L. (2004). *Combining Pattern Classifiers: Methods and Algorithms*. NY: Wiley.
- Laith, Abualigah and Ali, Diabat (2021). “Advances in Sine Cosine Algorithm: A comprehensive survey”. In: *Artificial Intelligence Review* 54, pp. 2567–2608.
- Laith, Abualigah, Ali, Diabat, Seyedali, Mirjalili, Mohamed, Abd Elaziz, and Amir H., Gandomi (2021). “The Arithmetic Optimization Algorithm”. In: *Computer Methods in Applied Mechanics and Engineering* 376.113609. URL: <https://doi.org/10.1016/j.cma.2020.113609>.
- Laith, Abualigah, Dalia, Yousri, Mohamed Abd, Elaziz, Ahmed A., Ewees, Mohammed A.A., Al-qaness, and Amir H., Gandomi (2021). “Aquila Optimizer: A novel meta-heuristic optimization algorithm”. In: *Computers and Industrial Engineering* 157.107250. URL: <https://doi.org/10.1016/j.cie.2021.107250>.
- Lan, Zhenzhong, Chen, Mingda, Goodman, Sebastian, Gimpel, Kevin, Sharma, Piyush, and Soricut, Radu (2019). “Albert: A lite bert for self-supervised learning of language representations”. In: *arXiv preprint arXiv:1909.11942*. URL: <https://arxiv.org/abs/1909.11942>.
- Latif, S., Rana, R., Qadir, J., and Epps, J. (2017). “Variational autoencoders for learning latent representations of speech emotion: A preliminary study”. In: *arXiv preprint arXiv:1712.08708*.

- LeCun, Yann, Boser, Bernhard, Denker, John, Henderson, Donnie, Howard, R., Hubbard, Wayne, and Jackel, Lawrence (1989). “Handwritten Digit Recognition with a Back-Propagation Network”. In: *Advances in Neural Information Processing Systems*. Ed. by D. Touretzky. Vol. 2. Morgan-Kaufmann. URL: <https://proceedings.neurips.cc/paper/1989/file/53c3bce66e43be4f209556518c2fcb54-Paper.pdf>.
- Lee, Sanghyun, Han, David K, and Ko, Hanseok (2020). “Fusion-ConvBERT: parallel convolution and BERT fusion for speech emotion recognition”. In: *Sensors* 20.22, p. 6688.
- (2021). “Multimodal emotion recognition fusion analysis adapting BERT with heterogeneous feature unification”. In: *IEEE Access* 9, pp. 94557–94572.
- Liu, Bin (2016). *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press, 2020.
- Liu, Yinhan, Ott, Myle, Goyal, Naman, Du, Jingfei, Joshi, Mandar, Chen, Danqi, Levy, Omer, Lewis, Mike, Zettlemoyer, Luke, and Stoyanov, Veselin (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. DOI: [10.48550/ARXIV.1907.11692](https://arxiv.org/abs/1907.11692). URL: <https://arxiv.org/abs/1907.11692>.
- Madhfar, M. A. H. and Al-Hagery, M. A. H. (2019). “Arabic Text Classification: A Comparative Approach Using a Big Dataset”. In: *2019 International Conference on Computer and Information Sciences (ICCIS)*, pp. 1–5. DOI: [10.1109/ICCISci.2019.8716479](https://doi.org/10.1109/ICCISci.2019.8716479).
- Malmasi, Shervin, Refaee, Eshrag, and Dras, Mark (2016). “Arabic Dialect Identification Using a Parallel Multidialectal Corpus”. In: *Computational Linguistics*. Ed. by Kôiti Hasida and Ayu Purwarianti. Singapore: Springer Singapore, pp. 35–53. ISBN: 978-981-10-0515-2.
- Manning, Christopher and Schütze, Hinrich (1999). *Foundations of statistical natural language processing*. MIT press.
- Martin, Louis, Muller, Benjamin, Suárez, Pedro Javier Ortiz, Dupont, Yoann, Romary, Laurent, Clergerie, Éric Villemonte de la, Seddah, Djamel, and Sagot, Benoit (2020). “CamemBERT: a Tasty French Language Model”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. URL: <https://www.aclweb.org/anthology/2020.acl-main.645>.

- Mataoui, Mhamed, Zelmami, Omar, and Boumechache, Madiha (2016). “A Proposed Lexicon-Based Sentiment Analysis Approach for the Vernacular Algerian Arabic”. In: *Research in Computing Science* 110, pp. 55–70. DOI: [10.13053/racs-110-1-5](https://doi.org/10.13053/racs-110-1-5).
- Meftouh, Karima, Harrat, Salima, Jamoussi, Salma, Abbas, Mourad, and Smaili, Kamel (2015). “Machine translation experiments on PADIC: A parallel Arabic dialect corpus”. In: *Proceedings of The 29th Pacific Asia Conference on Language, Information and Computation, shanghai, China*, pp. 26–34.
- Mikolov, Tomas, Chen, Kai, Corrado, Greg, and Dean, Jeffrey (2013). “Efficient estimation of word representations in vector space”. In: *arXiv preprint arXiv:1301.3781*.
- Mikolov, Tomas, Grave, Edouard, Bojanowski, Piotr, Puhersch, Christian, and Joulin, Armand (2018). “Advances in Pre-Training Distributed Word Representations”. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC)*. Miyazaki, Japan. DOI: <https://www.aclweb.org/anthology/L18-1008>.
- Mikolov, Tomas, Sutskever, Ilya, Chen, Kai, Corrado, Greg S, and Dean, Jeff (2013). “Distributed Representations of Words and Phrases and their Compositionality”. In: *Advances in Neural Information Processing Systems*. Ed. by C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger. Vol. 26. Curran Associates, Inc. URL: <https://proceedings.neurips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf>.
- Mikolov, Tomás, Chen, Kai, Corrado, Greg, and Dean, Jeffrey (2013). *Efficient Estimation of Word Representations in Vector Space*. Ed. by Yoshua Bengio and Yann LeCun. URL: <http://arxiv.org/abs/1301.3781>.
- Mohammed, Pasant, Eid, Yomna, Badawy, Mahmoud, and Hassan, Ahmed (2019). “Evaluation of Different Sarcasm Detection Models for Arabic News Headlines”. In: *International Conference on Advanced Intelligent Systems and Informatics*. Springer, pp. 418–426.
- Morrison, D. and De Silva, L. C. (2007). “Voting ensembles for spoken affect classification”. In: *Journal of Network and Computer Applications* 30.4, pp. 1356–1365.

- Morrison, D. and Silva, L. C. De (2007). “Voting ensembles for spoken affect classification”. In: *Journal of Network and Computer Applications* 30.4, pp. 1356–1365. DOI: [10.1016/j.jnca.2006.09.005](https://doi.org/10.1016/j.jnca.2006.09.005).
- Morrison, D., Wang, R., and Silva, L. C. De (2007). “Ensemble methods for spoken emotion recognition in call-centres”. In: *Speech Communication* 49.2, pp. 98–112. DOI: [10.1016/j.specom.2006.11.004](https://doi.org/10.1016/j.specom.2006.11.004).
- Moudjari, Leila and Akli-Astouati, Karima (2020). “An Experimental Study on Sentiment Classification of Algerian Dialect Texts”. In: *Procedia Computer Science* 176, pp. 1151–1159.
- Moudjari, Leila, Akli-Astouati, Karima, and Benamara, Farah (2020). “An Algerian Corpus and an Annotation Platform for Opinion and Emotion Analysis”. In: *Proceedings of The 12th Language Resources and Evaluation Conference*, pp. 1202–1210. URL: <https://www.aclweb.org/anthology/2020.lrec-1.151/>.
- Mozannar, Hussein, Maamary, Elie, El Hajal, Karl, and Hajj, Hazem (Aug. 2019). “Neural Arabic Question Answering”. In: *Proceedings of the Fourth Arabic Natural Language Processing Workshop*. Florence, Italy: Association for Computational Linguistics, pp. 108–118. DOI: [10.18653/v1/W19-4612](https://doi.org/10.18653/v1/W19-4612). URL: <https://aclanthology.org/W19-4612>.
- Muaad, Abdullah Y, Kumar, G Hemantha, Hanumanthappa, J, Benifa, JV Bibal, Mourya, M Naveen, Chola, Channabasava, Pramodha, M, and Bhairava, R (2022). “An effective approach for Arabic document classification using machine learning”. In: *Global Transitions Proceedings* 3.1, pp. 267–271.
- Mubarak, H., Darwish, K., and Magdy, W. (Aug. 2017). “Abusive language detection on Arabic social media”. In: *Proceedings of the first workshop on abusive language online*, pp. 52–56.
- Mulki, Hala, Haddad, Hatem, Bechikh Ali, Chedi, and Alshabani, Halima (2019). “L-HSAB: A Levantine Twitter Dataset for Hate Speech and Abusive Language”. In: DOI: [10.18653/v1/W19-3512](https://doi.org/10.18653/v1/W19-3512).
- Nabil, Mahmoud, Aly, Mohamed, and Atiya, Amir (2015). “Astd: Arabic sentiment tweets dataset”. In: *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 2515–2519.

- Napu, Novriyanto (Sept. 2018). “ENGLISH AND INDONESIAN NEWSPAPER HEADLINES: A COMPARATIVE STUDY OF LEXICAL FEATURES”. In: 2, pp. 103–116. DOI: [10.5281/zenodo.1419261](https://doi.org/10.5281/zenodo.1419261).
- Ngai, Hillary, Park, Yoona, Chen, John, and Parsapoor, Mahboobeh (2021). *Transformer-Based Models for Question Answering on COVID19*. DOI: [10.48550/ARXIV.2101.11432](https://doi.org/10.48550/ARXIV.2101.11432). URL: <https://arxiv.org/abs/2101.11432>.
- Ombabi, Abubakr H, Ouarda, Wael, and Alimi, Adel M (2020). “Deep learning CNN-LSTM framework for Arabic sentiment analysis using textual information shared in social networks”. In: *Social Network Analysis and Mining* 10.1, pp. 1–13.
- P.R. Cavalin R. Sabourin, C.Y. Suen (2013). “Dynamic selection approaches for multiple classifier systems”. In: *Neural Computing and Applications* 22, pp. 673–688.
- Pennington, Jeffrey, Socher, Richard, and Manning, Christopher (Oct. 2014). “GloVe: Global Vectors for Word Representation”. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543. DOI: [10.3115/v1/D14-1162](https://doi.org/10.3115/v1/D14-1162). URL: <https://www.aclweb.org/anthology/D14-1162>.
- Peters, Matthew, Neumann, Mark, Iyyer, Mohit, Gardner, Matt, Clark, Christopher, Lee, Kenton, and Zettlemoyer, Luke (June 2018). “Deep Contextualized Word Representations”. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 2227–2237. DOI: [10.18653/v1/N18-1202](https://doi.org/10.18653/v1/N18-1202). URL: <https://www.aclweb.org/anthology/N18-1202>.
- Plutchik, Robert (1984). “Emotions: A general psychoevolutionary theory”. In: *Approaches to emotion* 1984, pp. 197–219.
- Polignano, Marco, Basile, Pierpaolo, Gemmis, Marco de, Semeraro, Giovanni, and Basile, Valerio (Nov. 2019). “ALBERTO: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets”. In: *6th Italian Conference on Computational Linguistics, CLiC-it 2019*. Vol. 2481. URL: <http://ceur-ws.org/Vol-2481/paper57.pdf>.

- Radford, Alec, Kim, Jong Wook, Xu, Tao, Brockman, Greg, McLeavey, Christine, and Sutskever, Ilya (2022). “Robust speech recognition via large-scale weak supervision”. In: *arXiv preprint arXiv:2212.04356*.
- Radford, Alec, Wu, Jeffrey, Child, Rewon, Luan, David, Amodei, Dario, and Sutskever, Ilya (2019). “Language models are unsupervised multitask learners”. In: *Technical Report OpenAI 1*, pp. 1–24.
- Ravanelli, Mirco, Parcollet, Titouan, Plantinga, Peter, Rouhe, Aku, Cornell, Samuele, Lugosch, Loren, Subakan, Cem, Dawalatabad, Nauman, Heba, Abdelwahab, Zhong, Jianyuan, et al. (2021). “SpeechBrain: A general-purpose speech toolkit”. In: *arXiv preprint arXiv:2106.04624*.
- Rebiai, Zinedine, Andersen, Simon, Debrenne, Antoine, and Lafargue, Victor (2019). “SCIA at SemEval-2019 Task 3: sentiment analysis in textual conversations using deep learning”. In: *Proceedings of the 13th international workshop on semantic evaluation*, pp. 297–301.
- Rosenthal, Sara, Farra, Noura, and Nakov, Preslav (2017). “SemEval-Task 4: Sentiment Analysis in Twitter”. In: *Proceedings of the 11th International Workshop on Semantic Evaluation*. Association for Computational Linguistics, pp. 502–518. DOI: [10.18653/v1/S17-2088](https://doi.org/10.18653/v1/S17-2088).
- Ruder, Sebastian, Peters, Matthew E., Swayamdipta, Swabha, and Wolf, Thomas (June 2019). “Transfer Learning in Natural Language Processing”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 15–18. DOI: [10.18653/v1/N19-5004](https://doi.org/10.18653/v1/N19-5004). URL: <https://www.aclweb.org/anthology/N19-5004>.
- Sayadi, K., Liwicki, M., Ingold, R., and Bui, M. (2016). “Tunisian dialect and modern standard arabic dataset for sentiment analysis: Tunisian election context”. In: *Second International Conference on Arabic Computational Linguistics, ACLING*, pp. 35–53.
- Sayed, Mostafa, Salem, Rashed K, and Khder, Ayman E (2019). “A survey of Arabic text classification approaches”. In: *International Journal of Computer Applications in Technology* 59.3, pp. 236–251.

- Schuller, B. and Rigoll, G. (2006). “Timing levels in segment-based speech emotion recognition”. In: *Proc. INTERSPEECH 2006, Proc. Int. Conf. on Spoken Language Processing ICSLP*. Pittsburgh, USA.
- Schulz, Pamela (Apr. 2008). “Rougher than Usual Media Treatment: A Discourse Analysis of Media Reporting and Justice on Trial”. In.
- Schuster, Mike and Paliwal, Kuldeep K (1997). “Bidirectional recurrent neural networks”. In: *IEEE transactions on Signal Processing* 45.11, pp. 2673–2681.
- Seddah, D. et al. (July 2020). “Building a user-generated content north-african arabizi treebank: Tackling hell”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1139–1150.
- Seddah, Djamé, Essaidi, Farah, Fethi, Amal, Futeral, Matthieu, Muller, Benjamin, Suárez, Pedro Javier Ortiz, Sagot, Benoit, and Srivastava, Abhishek (2020). “Building a User-Generated Content North-African Arabizi Treebank: Tackling Hell”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 1139–1150. URL: <https://aclanthology.org/2020.acl-main.107/>.
- Shahin, I., Nassif, A. B., and Hamsa, S. (2019). “Emotion Recognition Using Hybrid Gaussian Mixture Model and Deep Neural Network”. In: *IEEE Access* 7, pp. 26777–26787. DOI: [10.1109/access.2019.2901352](https://doi.org/10.1109/access.2019.2901352).
- Shen, P., Changjun, Z., and Chen, X. (2011). “Automatic speech emotion recognition using support vector machine”. In: *Proceedings of 2011 International Conference on Electronic & Mechanical Engineering and Information Technology*. Vol. 2, pp. 621–625.
- Shon, Suwon, Ali, Ahmed, Samih, Younes, Mubarak, Hamdy, and Glass, James (2020). “ADI17: A Fine-Grained Arabic Dialect Identification Dataset”. In: *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8244–8248. DOI: [10.1109/ICASSP40776.2020.9052982](https://doi.org/10.1109/ICASSP40776.2020.9052982).
- Soumeur, Assia, Mokdadi, Mheni, Guessoum, Ahmed, and Daoud, Amina (2018). “Sentiment analysis of users on social networks: overcoming the challenge of the loose usages of the Algerian Dialect”. In: *Procedia computer science* 142, pp. 26–37.

- Srivastava, Nitish, Hinton, Geoffrey, Krizhevsky, Alex, Sutskever, Ilya, and Salakhutdinov, Ruslan (2014). “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1, pp. 1929–1958.
- Sun, Chen, Shrivastava, Abhinav, Singh, Saurabh, and Gupta, Abhinav (2017). “Revisiting unreasonable effectiveness of data in deep learning era”. In: *Proceedings of the IEEE international conference on computer vision*, pp. 843–852.
- Talafha, Bashar, Ali, Mohammad, Za’ter, Muhy Eddin, Seelawi, Haitham, Tuffaha, Ibraheem, Samir, Mostafa, Farhan, Wael, and Al-Natsheh, Hussein (Dec. 2020). “Multi-dialect Arabic BERT for Country-level Dialect Identification”. In: *Proceedings of the Fifth Arabic Natural Language Processing Workshop*. Barcelona, Spain (Online): Association for Computational Linguistics, pp. 111–118. URL: <https://www.aclweb.org/anthology/2020.wanlp-1.10>.
- Tobaili, T (Aug. 2016). “Arabizi identification in twitter data”. In: *Proceedings of the ACL 2016 Student Research Workshop*, pp. 51–57.
- Torfi, Amirsina, Shirvani, Rouzbeh A, Keneshloo, Yaser, Tavaf, Nader, and Fox, Edward A (2020). “Natural language processing advancements by deep learning: A survey”. In: *arXiv preprint arXiv:2003.01200*.
- Touileb, Samia (Nov. 2022). “NERDz: A Preliminary Dataset of Named Entities for Algerian”. In: *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Online only: Association for Computational Linguistics, pp. 95–101. URL: <https://aclanthology.org/2022.aacl-short.13>.
- Touileb, Samia and Barnes, Jeremy (2021). “The interplay between language similarity and script on a novel multi-layer Algerian dialect corpus”. In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 3700–3712. URL: <https://aclanthology.org/2021.findings-acl.324/>.
- Vaswani, Ashish, Shazeer, Noam, Parmar, Niki, Uszkoreit, Jakob, Jones, Llion, Gomez, Aidan N, Kaiser, ukasz, and Polosukhin, Illia (2017a). “Attention is All you Need”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach,

- R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- (2017b). “Attention is All you Need”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Ed. by Guyon et al. Vol. 30. URL: <https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf>.
- Vilalta, R. and Drissi, Y. (2002). “A perspective view and survey of meta-learning”. In: *Artificial Intelligence Review* 18(2), pp. 77–95.
- Wang, T.Y. and Chiang, H.M. (2011). “Solving multi-label text categorization problem using support vector machine approach with membership function”. In: *Neurocomputing* 74, pp. 3692–3689.
- Wolf, Thomas, Debut, Lysandre, Sanh, Victor, Chaumond, Julien, Delangue, Clement, Moi, Anthony, Cistac, Pierric, Rault, Tim, Louf, Remi, Funtowicz, Morgan, Davison, Joe, Shleifer, Sam, Platen, Patrick von, Ma, Clara, Jernite, Yacine, Plu, Julien, Xu, Canwen, Le Scao, Teven, Gugger, Sylvain, Drame, Mariama, Lhoest, Quentin, and Rush, Alexander (Oct. 2020). “Transformers: State-of-the-Art Natural Language Processing”. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online, pp. 38–45. DOI: [10.18653/v1/2020.emnlp-demos.6](https://doi.org/10.18653/v1/2020.emnlp-demos.6). URL: <https://www.aclweb.org/anthology/2020.emnlp-demos.6>.
- Wu, Yonghui, Schuster, Mike, Chen, Zhifeng, Le, Quoc V, Norouzi, Mohammad, Macherey, Wolfgang, Krikun, Maxim, Cao, Yuan, Gao, Qin, Macherey, Klaus, et al. (2016). “Google’s neural machine translation system: Bridging the gap between human and machine translation”. In: *arXiv preprint arXiv:1609.08144*. URL: <https://arxiv.org/abs/1609.08144>.
- Xu, Hu, Liu, Bing, Shu, Lei, and Yu, Philip (June 2019). “BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis”. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 2324–2335. DOI: [10.18653/v1/N19-1242](https://doi.org/10.18653/v1/N19-1242). URL: <https://aclanthology.org/N19-1242>.

- Y.Hifny and A.Ali (May 2019). “Efficient arabic emotion recognition using deep neural networks”. In: *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6710–6714.
- Y.Li, T.Zhao, and T.Kawahara (Sept. 2019). “Improved end-to-end speech emotion recognition using self attention mechanism and multitask learning”. In: *Interspeech*, pp. 2803–2807.
- Yahia Cherif, Raoudha, Moussaoui, Abdelouahab, Frahta, Nabila, and Berrimi, Mohamed (2021). “Effective speech emotion recognition using deep learning approaches for Algerian dialect”. In: *2021 International Conference of Women in Data Science at Taif University (WiDSTaif)*. IEEE, pp. 1–6.
- Yang, Zhilin, Dai, Zihang, Yang, Yiming, Carbonell, Jaime, Salakhutdinov, Russ R, and Le, Quoc V (2019). “Xlnet: Generalized autoregressive pretraining for language understanding”. In: *Advances in neural information processing systems* 32.
- Yang, Zichao, Yang, Diyi, Dyer, Chris, He, Xiaodong, Smola, Alex, and Hovy, Eduard (2016). “Hierarchical attention networks for document classification”. In: *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 1480–1489.
- Ykhlef, Fay., Derbal, A., Benzaba, W., Boutaleb, R., Bouchaffra, D., Meraoubi, H., and Ykhlef, Far. (2019). “Towards Building an Emotional Speech Corpus of Algerian Dialect: Criteria and Preliminary Assessment Results”. In: *2019 International Conference on Advanced Electrical Engineering (ICAEE)*, pp. 1–6. DOI: [10.1109/ICAEE47123.2019.9014808](https://doi.org/10.1109/ICAEE47123.2019.9014808).
- Younes, Jihene, Souissi, Emna, Achour, Hadhemi, and Ferchichi, Ahmed (2020). “Language resources for Maghrebi Arabic dialects’ NLP: a survey”. In: *Language Resources and Evaluation* 54.4, pp. 1079–1142. URL: <https://link.springer.com/article/10.1007/s10579-020-09490-9>.
- Yuan, Ye, Xun, Guangxu, Suo, Qiuling, Jia, Kebin, and Zhang, Aidong (2017). “Wave2vec: Learning deep representations for biosignals”. In: *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, pp. 1159–1164.

- Zaidan, Omar F. and Callison-Burch, Chris (Mar. 2014). “Arabic Dialect Identification”. In: *Computational Linguistics* 40.1, pp. 171–202. DOI: [10.1162/COLI_a_00169](https://doi.org/10.1162/COLI_a_00169). URL: <https://aclanthology.org/J14-1006>.
- Zantout, R., Klaylat, S., Hamandi, L., and Osman, Z. (Mar. 2019). “Ensemble Models for Enhancement of an Arabic Speech Emotion Recognition System”. In: *Future of Information and Communication Conference*, pp. 174–187.
- Zeng, Jichuan, Li, Jing, Song, Yan, Gao, Cuiyun, Lyu, Michael R, and King, Irwin (2018). “Topic memory networks for short text classification”. In: *arXiv preprint arXiv:1809.03664*.
- Zhan, Justin and Dahal, Binay (2017). “Using deep learning for short text understanding”. In: *Journal of Big Data* 4.1, p. 34.
- Zhang, Lei, Ghosh, Riddhiman, Dekhil, Mohamed, Hsu, Meichun, and Liu, Bing (Jan. 2011). “Combining Lexicon-based and Learning-based Methods for Twitter Sentiment Analysis”. In: *HP Laboratories, Report No: HPL-2011-89, 2011*, pp. 1–8.
- Zhao, J., Mao, X., and Chen, L. (2019). “Speech emotion recognition using deep 1D & 2D CNN LSTM networks”. In: *Biomedical Signal Processing and Control* 47, pp. 312–323.
- Zhou, Peng, Shi, Wei, Tian, Jun, Qi, Zhenyu, Li, Bingchen, Hao, Hongwei, and Xu, Bo (Aug. 2016). “Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification”. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 207–212. DOI: [10.18653/v1/P16-2034](https://doi.org/10.18653/v1/P16-2034). URL: <https://aclanthology.org/P16-2034>.