



THE UNIVERSITY *of* EDINBURGH

This thesis has been submitted in fulfilment of the requirements for a postgraduate degree (e. g. PhD, MPhil, DClinPsychol) at the University of Edinburgh. Please note the following terms and conditions of use:

- This work is protected by copyright and other intellectual property rights, which are retained by the thesis author, unless otherwise stated.
- A copy can be downloaded for personal non-commercial research or study, without prior permission or charge.
- This thesis cannot be reproduced or quoted extensively from without first obtaining permission in writing from the author.
- The content must not be changed in any way or sold commercially in any format or medium without the formal permission of the author.
- When referring to this work, full bibliographic details including the author, title, awarding institution and date of the thesis must be given.

Arabic Sarcasm Detection

Ibrahim Abu Farha



Doctor of Philosophy
Institute for Language, Cognition and Computation
School of Informatics
University of Edinburgh
2023

Abstract

Sarcasm is a form of verbal irony that is often used to express ridicule or contempt. When using sarcasm, a speaker expresses their opinion in an indirect way, where the literal meaning is different from the intended one. Additionally, sarcasm is a sociolinguistic tool that people use to express themselves and it reflects their cultural and social background. Sarcasm detection refers to the process of automatically and computationally identifying whether a piece of text is sarcastic. This has been well studied in the context of English, but Arabic lags behind. In this thesis, we try to fill in the gaps in the research on Arabic sarcasm detection.

First, we start by exploring approaches to create an Arabic sarcasm dataset. We create ArSarcasm dataset through the re-annotation of existing sentiment analysis datasets. These labels represent perceived sarcasm as the labels reflect the annotators' perception. The analysis shows that sarcasm is prominent in the used sentiment datasets, with 16% of the sentences being sarcastic. Our experiments show that sarcasm is disruptive for sentiment analysers. Analysis shows that annotating subjective content can be challenging and prone to biases.

Second, to mitigate the issues and fallbacks of sarcasm data collection approaches, we propose to collect sarcasm datasets by asking people to label their words, which is referred to as intended sarcasm. The resulting dataset, which is first-party annotated, would have more reliable and trustworthy labels and does not have the issues of third-party annotated data.

Next, we test state-of-the-art machine learning models on the newly created datasets. Those experiments provide a benchmark for these datasets. The experiments show that intended sarcasm detection is more challenging than perceived sarcasm detection. Also, the experiments show that monolingual Arabic language models, which include dialects in their pre-training data, perform better on the sarcasm detection task. Additionally, we provide the details of shared tasks that utilise the new datasets.

Finally, we provide an in-depth error analysis comparing humans' performance in sarcasm detection against the performance of state-of-the-art models. Our analysis confirms that sarcasm is challenging for both humans and machines. We also highlight the features and patterns used to express sarcasm, such as idioms and proverbs. When extending the analysis to focus on Arabic dialects, we found that dialect familiarity affects how Arabic speakers understand and interpret sarcasm. Arabic speakers were better able to detect sarcasm expressed in their dialect or one they were familiar with.

Acknowledgements

I feel incredibly fortunate to have had the opportunity to pursue my PhD at the University of Edinburgh. My deepest gratitude goes to my supervisor, Walid Magdy—thank you, Walid! Throughout the years, you have been my supervisor, mentor, friend, and brother. Your unwavering guidance and support have been the driving force behind my accomplishments, and I will forever be grateful to you.

I would also like to extend my thanks to Bonnie Webber, Muhammad Abdul-Mageed, Kareem Darwish, and Steve Wilson for their valuable feedback and suggestions during my annual reviews. Also, many thanks to my examiners, Nizar Habash and Adam Lopez. Thank you for your insightful feedback, deep discussions and for making the viva smooth and enjoyable.

To my family—my mother, father, sister, fiancé, and brother—I am grateful for your unconditional love and support. A special mention goes to my brother, Yazan, for his constant encouragement, guidance, and dedication in always being there for me. My journey would not have been the same without you.

I am grateful to my friends, both near and far: Ahmad, Youssef, Maysara, Youcef, Mohammad, Yazan, Julie-Anne, and Karim. Your presence, especially during the challenging lockdowns, was invaluable. I also want to acknowledge my friends in the SMASH group: Amr, Silviu, Steve, Dilara, Lushi, Bjorn, Xue, and Sabyasachi. Also, I wish to extend my heartfelt thanks to my friends in Edinburgh: Youssef, Isa, Hadi, Monzer, Erik, Matus, Parag, Eric, Asif, Micky, Alex, Dishan, and Razvan.

Additionally, I want to thank the British Council and HESPAL scholarship, who funded my MSc at the same university, which enabled me to pursue my PhD later.

Finally, I would like to express my thanks and gratitude to Dstl and Alan Turing Institute, who funded my PhD Journey.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(Ibrahim Abu Farha)

Contents

1	Introduction	1
1.1	Overview	1
1.2	Motivation	4
1.3	Thesis statement	5
1.4	Contributions	5
1.5	Thesis Structure	7
1.6	Thesis Outcomes	8
1.6.1	List of Publications	8
1.6.2	Released Datasets	9
2	Background	11
2.1	Arabic Language	11
2.2	Sarcasm	13
2.2.1	Definitions	13
2.2.2	Socio-linguistic Variables	14
2.2.3	Intended vs Perceived Sarcasm	15
2.2.4	Arabic Sarcasm	15
2.3	Arabic Natural Language Processing	18
2.3.1	Arabic Subjectivity and Sentiment Analysis	19
2.3.2	Arabic Language Models	23
2.4	Computational Sarcasm Analysis	24
2.4.1	English Sarcasm Detection	24
2.4.2	Arabic Sarcasm Detection	28
2.4.3	Sarcasm and Irony Detection in Other Languages	29
2.5	Research Gap	30

3	Annotating Arabic Sentiment Datasets for Sarcasm	33
3.1	Overview	34
3.2	ArSarcasm Dataset	35
3.2.1	Resources	35
3.2.2	Annotation	36
3.3	Statistics and Analysis	37
3.3.1	Dataset Statistics	37
3.3.2	Sentiment in Sarcasm	38
3.3.3	Annotation Subjectivity	39
3.4	Effect of Sarcasm on Sentiment Analysis	42
3.5	Arabic Sarcasm Shared Task	42
3.5.1	ArSarcasm-v2 Dataset	43
3.5.2	Shared Task Details	45
3.5.3	Tasks Description	45
3.5.4	Discussion	50
3.6	Summary	50
4	Intended Sarcasm Dataset	53
4.1	Overview	53
4.2	Dataset	55
4.2.1	Overview	55
4.2.2	Data Collection	55
4.2.3	Quality Control	57
4.2.4	Limitations	58
4.3	iSarcasmEval Details	59
4.3.1	Task Description	59
4.3.2	iSarcasmEval Arabic Dataset	59
4.3.3	Evaluation Metrics	60
4.4	Participating Teams	61
4.4.1	Overview	61
4.4.2	Subtask A (Sarcasm Detection) - Arabic	61
4.4.3	Subtask C (Pairwise Sarcasm Identification) - Arabic	61
4.4.4	Discussion	63
4.5	Summary	64

5	Arabic Sarcasm Detection, Modelling and Experiments	65
5.1	Overview	66
5.2	Experimental Setup	67
5.2.1	Datasets	67
5.2.2	Models	67
5.2.3	Hyper-parameters and evaluation	70
5.3	Results	70
5.4	Discussion	71
5.4.1	Computational Cost	73
5.5	Summary	73
6	Humans and Sarcasm Detection	75
6.1	Human vs Machine	76
6.1.1	Analysis of Sarcasm Detection	77
6.1.2	Methodology	78
6.1.3	Results and Analysis	79
6.1.4	Discussion	84
6.1.5	Recommendations	85
6.2	Sarcasm and Arabic Dialects	86
6.2.1	Related Work	87
6.2.2	Methodology	88
6.2.3	Results and Analysis	89
6.2.4	Discussion	96
6.3	Summary	97
7	Conclusions & Future Work	99
7.1	Conclusions	99
7.2	Implications	101
7.3	Limitations & Future Directions	101
7.4	Impact	103
	Bibliography	105
	Arabic Bibliography	127
A	Annotation Guidelines	129

B Arabic iSarcasmEval Data Collection Instructions	131
C Results on Sentiment Analysis Task	133
D ChatGPT Experiments	135

List of Figures

2.1	The geographic distribution of Arabic dialects.	13
2.2	Snippets from movies containing sentences that are used as sarcastic responses on Arabic social media.	17
3.1	Ratio of sarcasm over the dialects.	38
3.2	Sentiment distribution over the sarcastic tweets.	39
3.3	The change in sentiment labels between the original and new annotation. The labels above the charts are the original labels.	41
4.1	A screenshot of the data collection form.	56
5.1	Time (in seconds) needed to fine-tune each model for one epoch for the sarcasm detection task (ArSarcasm-v2 dataset).	74
6.1	Prediction distribution for task A (Arabic).	80
6.2	Performance of human vs machine on task C.	83
6.3	Sarcasm detection performance ($F_1^{\text{sarcastic}}$) of different dialects speakers on each dialect. Original dialect labels were used. Annotation counts are indicated in brackets.	90
6.4	Annotators' sarcasm detection performance ($F_1^{\text{sarcastic}}$) when the dialect is <i>known</i> to them. Annotation counts are indicated in brackets.	91
6.5	Annotators' sarcasm detection performance ($F_1^{\text{sarcastic}}$) when the dialect is <i>unknown</i> to them. Annotation counts are indicated in brackets.	91
6.6	Annotators' sarcasm detection performance ($F_1^{\text{sarcastic}}$) when they <i>correctly</i> identified the dialect. Annotation counts are indicated in brackets.	92
6.7	Annotators' sarcasm detection performance ($F_1^{\text{sarcastic}}$) when they <i>incorrectly</i> identified the dialect. Annotation counts are indicated in brackets.	92

6.8	Sarcasm detection performance ($F_1^{\text{sarcastic}}$) based on the annotators' gender. Annotation counts are indicated in brackets.	93
6.9	Dialect identification accuracy of annotators speaking different dialects. Annotation counts are indicated in brackets.	94
6.10	Assigned dialect labels vs the original ones. Annotation counts are indicated in brackets.	94
6.11	Egyptian (Nile) speakers' dialect identification performance.	94
6.12	Levantine speakers' dialect identification performance.	95
6.13	Gulf speakers' dialect identification performance.	95
6.14	Maghrebi speakers' dialect identification performance.	95
1.1	A sample of the guidelines provided to the annotators.	130
2.1	A snippet of the instructions used to collect the Arabic iSarcasmEval dataset.	132
4.1	Incorrect ChatGPT answer for a sarcastic example.	136
4.2	Correct ChatGPT answer for a sarcastic example. Incorrect explanation.	136
4.3	Correct ChatGPT answer for a sarcastic example. Incorrect explanation.	137
4.4	Inorrect ChatGPT answer for a non-sarcastic example.	137
4.5	Correct ChatGPT answer for a non-sarcastic example. Incorrect explanation and translation.	138
4.6	Incorrect ChatGPT answer for a non-sarcastic example.	138

List of Tables

2.1	Examples of some sarcastic sentences in different dialects. LT stands for literal translation. IM stands for intended meaning.	18
2.2	Summary of previous approaches in Arabic sentiment Analysis. The polarities are positive (POS), negative (NEG), or neutral (NEU). . . .	22
2.3	Summary of sarcasm detection approaches in the literature.	31
3.1	ASTD statistics.	35
3.2	SemEval 2017 Task 4-A dataset statistics.	36
3.3	Dataset statistics for sarcasm and sentiment over the dialects.	38
3.4	Examples of some sarcastic tweets from different dialects. LT stands for literal translation. IM stands for intended meaning.	40
3.5	Examples of some tweets that have their labels changed.	41
3.6	Mazajak’s performance on sarcastic and non-sarcastic tweets. The references are the original and the new sentiment labels.	42
3.7	Statistics of ArSarcasm-v2 dataset showing the distribution of sarcasm and sentiment over the dialects.	44
3.8	Statistics of training and testing sets, showing the number of examples for both sarcasm detection and sentiment analysis tasks.	45
3.9	The list of participating teams who provided their affiliation details. . .	46
3.10	Results achieved by participants in subtask 1 (sarcasm detection). The main metric is the F1-score of the sarcastic class ($F_1^{\text{sarcastic}}$).	48
3.11	Results achieved by participants in subtask 2 (sentiment analysis). The main metric is the macro average of the F1-scores of the positive and negative classes (F_1^{PN}).	49
4.1	Examples of sarcastic sentences from our dataset along with the rephrase that authors gave that convey the same meaning non-sarcastically and their explanation of why their sentence is sarcastic.	57

4.2	Statistics for the Arabic training set, and test sets for subtasks A and C, as discussed in Section 4.3.2.	60
4.3	Distribution of the Arabic dataset over the dialects.	60
4.4	Subtask A (Arabic) results in descending order according to the main metric ($F_1^{\text{sarcastic}}$). The table shows the teams’ names, rank, affiliation, and score.	62
4.5	Subtask C (Arabic) results in descending order according to the main metric (accuracy). The table shows the teams’ names, ranks, affiliations, and scores.	63
5.1	Summary of the details of each model. The table includes details about the model variant, architecture, training data size, training data nature, number of parameters. In text nature types are: tweets (T), Wikipedia (W), news (N), OSCAR corpus (OC), Common Crawl (CC), subtitles (S), and books (B). (CS) stands for code-switching, and (*) indicates that Farasa segmentation is applied to the text. Finally, (N/A) indicates that information is not available.	69
5.2	Results achieved by all models on ArSarcasm-v2 and iSarcasmEval datasets. The table includes precision, recall, and $F_1^{\text{sarcastic}}$ scores. . .	71
6.1	Results for humans with majority voting, humans individually, and the top performing system for both the sarcasm detection (task A) and pairwise identification (task C). $F_1^{\text{sarcastic}}$ is the $F1$ score for the sarcastic class, the official metric used in the shared task.	79
6.2	Cohen’s kappa agreement between the human (majority vote) annotations and the predictions from the top performing system in the respective task.	80
6.3	Arabic non-sarcastic examples.	81
6.4	Arabic sarcastic examples. LT stands for literal translation. IM stands for intended meaning.	82
6.5	Arabic pairs of sarcastic texts and their non-sarcastic rephrases.	83
6.6	Sarcasm themes among sarcastic Arabic sentences. N : number of examples, H_{Er} : human error, and M_{Er} : machine error	84
6.7	Distribution of the Arabic iSarcasmEval test set over the dialects. . . .	88

6.8	Annotators' details. The table shows the number of annotators who speak a specific dialect as a mother tongue and the number of annotators who mentioned that they know a specific dialect.	89
6.9	Sarcasm detection performance ($F_1^{\text{sarcastic}}$) of speakers of different dialects.	90
3.1	Results achieved by all models described in Chapter 5 on the sentiment analysis task in ArSarcasm-v2 dataset.	133

Chapter 1

Introduction

1.1 Overview

People express themselves through their words, either spoken or written. In the current era, social media platforms skyrocketed in popularity. People from all over the globe use social media platforms such as Facebook, Twitter, Reddit, etc., to express their ideas and opinions. In some cases, these platforms were the main arena to share ideas, express opinions, raise awareness, and criticise policies.

Social media has the advantage of being personal, free, and spontaneous. The abundance of online data and how accessible it is, made it the focus of natural language processing (NLP) researchers. Researchers started working on different tasks to provide an understanding of what people are expressing; such tasks include opinion mining, sentiment analysis, stance detection, and many others. Most of this research was focused on English, while other languages lagged behind. In the last two decades, Arabic NLP research started following this trend. Most of the Arabic NLP work focused on either core NLP tasks such as morphological analysis or applications such as sentiment analysis. Sentiment Analysis (SA) can be defined as the process of analysing and identifying the polarity/sentiment expressed in a piece of text, which can be from different sources such as social media posts or product reviews (Liu, 2012). SA is a crucial tool in analysing subjective content and helping to understand people's opinions/attitudes. For example, many companies rely on products' reviews in order to assess and readjust their marketing and planning strategies; in such a scenario SA would help save time and effort in analysing these reviews. In the early 2010s, the increase of Arabic web content, particularly on social media, and the transformative political developments in the Middle East attracted more interest to Arabic NLP applications,

including SA. However, when using social media, people tend to express their ideas in informal ways. In some cases, people resort to **sarcasm** as an indirect expression tool.

Sarcasm is a form of verbal irony that is often used to express ridicule or contempt. Sarcasm is usually correlated with expressing an opinion in an indirect way where there would be a discrepancy between the literal and intended meaning of an utterance (Wilson, 2006). The sentence *“I love being ignored”* is an example of sarcasm, where there is a discrepancy between the positive surface meaning and the negative implied one. The presence of sarcasm on social media imposes many challenges on computational systems harnessing this data. Examples of such systems are sentiment analysers, which are used to analyse how people feel towards a specific topic or event. Moderation and content filtering systems also can be affected. The importance of sarcasm and its effects attracted the attention of natural language processing (NLP) researchers. In the context of Arabic and Arabic NLP specifically, the progress is slower than in other languages, especially English. Arabic NLP researchers started filling the gaps over the last two decades.

Sarcasm falls under the umbrella of subjective content. The main challenge when working on such topics is that it is hard to define what is the ground truth. Subjective content, as the name implies, depends on whom you would ask. Opinions and beliefs would affect the perception of such content. Sentiment analysis suffers from this issue as people having different beliefs would have different, sometimes conflicting sentiments/feelings towards a piece of text. Sarcasm adds an additional challenge for systems utilising subjective content. In a sarcastic utterance, the meaning is convoluted and implied, making it challenging to determine the intended meaning. Most of the previous work on sarcasm detection and analysis focused on English. Other languages, such as Arabic, lagged behind.

Multi-word expressions (MWEs) such as idioms have been a challenge for machine translation (Carpuat and Diab, 2010; Zaninello and Birch, 2020; Baziotis et al., 2023). MWEs are a group of words whose linguistic behaviour is different from the individual component words (Van de Cruys and Villada Moirón, 2007). This challenge is also manifested in the presence of sarcasm, where the intended meaning of a sentence is different from the literal one. In this case, MT systems would struggle to produce a translation conveying the intended meaning in the target language.

Additionally, sarcasm has social dimensions that can be important to social and sociolinguistic scientists. Jorgensen (1996) showed that sarcasm is generally used to express complaints. However, it is usually used in a way that acts as a face-saving

function, making the speaker appear less rude. [Larson et al. \(2019\)](#) showed that sarcasm can be used to promote positive relationships. Work on sarcasm detection can help tackle the aforementioned challenges and provide a tool for computational social analysis of this linguistic phenomenon.

Consequently, detecting sarcasm and analysing it is of utmost importance. Previous work on sarcasm detection was mainly in two directions: creating resources and datasets and building detection models. However, this was mainly on English. Arabic sarcasm recently started receiving more attention, and the researchers followed a similar trajectory to the one that happened in English Sarcasm. However, there is still a gap in the research on Arabic sarcasm that needs to be filled.

1.2 Motivation

As mentioned previously, sarcasm can affect various NLP systems, such as sentiment analysers, machine translation systems, and content moderation systems. Considering how disruptive sarcasm can be for these systems, it is crucial to better understand sarcasm and how to detect it. Also, considering the social dimensions of sarcasm, it is necessary to create tools that can assist the computational analysis of sarcastic language. In the context of Arabic NLP, up to the start of this work, there were a few works on Arabic sarcasm. One of the early works is the work of [Karoui et al. \(2017\)](#), who explored Arabic irony/sarcasm detection. Other works include the shared task on irony/sarcasm detection ([Ghanem et al., 2019](#)) along with the participants' submissions and a dialectal sarcasm dataset by [Abbes et al. \(2020\)](#). These few works show that Arabic sarcasm is still under-studied and that a gap needs to be filled. In this work, we hypothesise the following:

- Sarcasm is omnipresent in Arabic social media across multiple Arabic dialects.
- Arabic sarcasm can be disruptive for sentiment analysis systems and would reduce their performance.
- Existing work on Arabic sarcasm is very limited and there should be more effort to create resources and systems to detect sarcasm.
- Arabic sarcasm on social media usually contains some features that help identify it.

Based on the previous hypotheses and throughout this thesis, we answer the following research questions:

- **RQ1:** Is sarcasm used across Arabic social media? If so, how prominent is it?
- **RQ2:** How does sarcasm affect the performance of sentiment analysers?
- **RQ3:** How to collect and reliably annotate a dataset for Arabic sarcasm detection?
- **RQ4:** What is the performance of the existing NLP approaches on Arabic sarcasm detection and how to improve it?
- **RQ5:** How do Arabic speakers perform on sarcasm detection and how does dialectal variation affect that?

1.3 Thesis statement

This thesis investigates the challenges and approaches to Arabic sarcasm detection. It provides an overview of Arabic sarcasm and explores how to better define, model, and collect sarcastic sentences. Through exploring various data collection approaches, it is found that sarcasm is extremely subjective and what is sarcastic depends on the person's interpretation. This thesis also shows the disruptive nature of sarcasm on sentiment analysis. Empirical analysis highlights the superior performance of monolingual Arabic language models incorporating dialects in sarcasm detection tasks. Through a comprehensive analysis of humans' and machines' performance, the thesis underscores the complexities of sarcasm comprehension in Arabic and sheds light on the linguistic and cultural influences that affect its perception.

1.4 Contributions

Over the course of this thesis, we answered the previous research questions and achieved the following contributions:

- **Creating the first publicly available Arabic Sarcasm dataset.**

Up to the start of this project, none of the previous datasets was publicly available. In this work, we created the first publicly available Arabic sarcasm dataset through the re-annotation of previous Arabic sentiment datasets. In addition to creating a new dataset, the availability of sentiment labels made it possible to analyse the effect of sarcasm on sentiment analysers. We found that sarcasm is prevalent in a significant portion of sentiment datasets, and it has a notable negative effect on the performance of sentiment analysers. This contribution addresses **RQ1** and **RQ2**.

- **Creating and publicly sharing three Arabic sarcasm datasets.**

In our effort to answer **RQ3**, we explored various approaches to creating sarcasm datasets, aiming to overcome the limitations of previous methods. In our work, we explored collecting both third-party and first-party annotations. We found that the main challenge is defining what is sarcastic. Since sarcasm is extremely subjective, different data collection methods would reflect different aspects of sarcasm. The resulting datasets enriched the resources available for Arabic sarcasm detection.

- **Creating and publicly sharing the first first-party annotated sarcasm dataset.**

In our effort to answer **RQ3**, we created an Arabic sarcasm dataset through asking people to provide sarcastic sentences, i.e. the labels are first-party. Consequently, the resulting dataset represents intended sarcasm as it captures the author's/speaker's intentions. This approach proved to be challenging and expensive as it requires many measures to maintain high-quality data. The resulting dataset is the first of its kind for Arabic and represents an important resource for sarcasm detection and analysis.

- **Bringing the attention of the Arabic NLP community to the sarcasm detection task. This was done through organising two shared tasks, one of which is at SemEval 2022¹.**

Through this PhD, we organised two shared tasks. The first was part of the Arabic NLP workshop and targeted Arabic sarcasm detection and sentiment analysis². The other task was part of SemEval 2022, namely iSarcasmEval³. Both shared tasks saw the participation of a large number of participants, which helped bring the attention of the Arabic NLP community to the sarcasm detection task. In addition to providing new datasets, the shared tasks provide a plethora of analysis and experiments of different models through the submissions of participating teams, which contribute to the answer of **RQ4**.

- **Benchmarking state-of-the-art models on Arabic sarcasm detection.**

To assess the effectiveness of different state-of-the-art models on sarcasm detection, we conducted comprehensive evaluations using the new datasets. Through these benchmarking experiments, we gained a deeper understanding of the strengths and weaknesses of various models, providing valuable insights into effective approaches for sarcasm detection. Generally, we found that the best models are monolingual Arabic transformer-based models that included dialectal Arabic in their pre-training data. These experiments and analyses contribute to the answer of **RQ4**.

- **An in-depth error analysis of the challenges of sarcasm for humans and machine models.**

¹<https://semeval.github.io/SemEval2022/tasks.html>

²<https://sites.google.com/view/ar-sarcasm-sentiment-detection/>

³<https://sites.google.com/view/semeval2022-isarcasmeval>

In addition to the datasets and experiments, we performed a thorough error analysis, comparing the performance of humans and state-of-the-art machine learning models in sarcasm detection. We provide both quantitative and qualitative analysis, which show the limitations of current models and the features that make sarcasm challenging. The analyses show that sarcasm detection is challenging for both humans and machines. The lack of contextual information was the main cause of incorrect predictions. The findings of these analyses contribute to the answer of **RQ5** through providing directions for future works.

- **Analysing the effect of dialectal variation on the process of data annotation.**

Dialectal variety is one of the main challenges when working on Arabic NLP. In our attempt to answer **RQ5**, we analysed how dialectal variation affects the data annotation process through the lens of sarcasm detection. The analysis shows that dialect familiarity can affect how people understand sarcasm.

1.5 Thesis Structure

The thesis chapters are organised as follows:

Chapter 2 provides an overview of background information covering Arabic and the challenges of working on Arabic NLP. It also covers sarcasm from a linguistic point of view, its types, and the factors that affect it. Additionally, we provide a general overview of Arabic NLP, Arabic subjectivity and sentiment analysis, and Arabic language models. Finally, a summary of related works on Arabic and English sarcasm detection is included.

Chapter 3 details the effort to create the ArSarcasm dataset. It provides the resources used, the annotation process, and in-depth analyses and discussions. The chapter provides a discussion of the issues arising when annotating subjective content. It also includes the details of the shared task on sarcasm and sentiment detection in Arabic along with the details of the participating submissions.

Chapter 4 discusses the issues of the previous approaches used and discussed in Chapter 3 and proposes a new approach to create sarcasm datasets. The chapter provides the process of creating the Arabic iSarcasmEval dataset and the details of the related shared task. It also provides the details of the top participating submissions.

Chapter 5 provides a set of experiments and analyses of the performance of state-of-the-art models on the Arabic sarcasm detection task. The chapter provides a com-

parison between the performance on perceived sarcasm (ArSarcasm-v2 dataset) and intended sarcasm (iSarcasmEval dataset).

Chapter 6 provides an in-depth analysis of the performance of state-of-the-art machine learning models and compares them to human performance on the sarcasm detection task. The chapter provides details of what makes sarcasm challenging for both humans and machine learning models. It also includes an analysis of how sarcasm is understood among Arabic speakers speaking different dialects.

Chapter 7 summarises the findings and discussion from previous chapters. It also provides conclusions and recommendations for future research on Arabic sarcasm.

1.6 Thesis Outcomes

During the PhD, we achieved the following outcomes, which are divided into publications and released datasets.

1.6.1 List of Publications

- Sarcasm related papers:

- 1- Ibrahim Abu Farha, Steven Wilson, Silviu Oprea, and Walid Magdy. 2022. Sarcasm Detection is Way Too Easy! An Empirical Comparison of Human and Machine Sarcasm Detection. In Findings of the Association for Computational Linguistics: EMNLP 2022, pages 5284–5295, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- 2- Ibrahim Abu Farha and Walid Magdy. 2022. The Effect of Arabic Dialect Familiarity on Data Annotation. In Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP), pages 399–408, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- 3- Ibrahim Abu Farha, Silviu Vlad Oprea, Steven Wilson, and Walid Magdy. 2022. SemEval-2022 Task 6: iSarcasmEval, Intended Sarcasm Detection in English and Arabic. In Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 802–814, Seattle, United States. Association for Computational Linguistics.

- 4- Ibrahim Abu Farha, Wajdi Zaghouani, 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, pages 296–305, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
 - 5- Ibrahim Abu Farha and Walid Magdy. 2021. Benchmarking Transformer-based Language Models for Arabic Sentiment and Sarcasm Detection. In Proceedings of the Sixth Arabic Natural Language Processing Workshop, pages 21–31, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
 - 6- Ibrahim Abu Farha and Walid Magdy. 2021. A Comparative Study of Effective Approaches for Arabic Sentiment Analysis. *Information Processing & Management*, 58(2):102438.
 - 7- Ibrahim Abu Farha and Walid Magdy. 2020. From Arabic Sentiment Analysis to Sarcasm Detection: The ArSarcasm Dataset. In Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, pages 32–39, Marseille, France. European Language Resource Association.
- Unrelated papers:
 - 1- Ibrahim Abu Farha and Walid Magdy. 2020. Multitask Learning for Arabic Offensive Language and Hate-Speech Detection. In Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, pages 86–90, Marseille, France. European Language Resource Association.

1.6.2 Released Datasets

- **ArSarcasm dataset**
 - The dataset contains 10,547 tweets annotated for sarcasm, sentiment, and dialect.
 - The first publicly available Arabic sarcasm dataset
 - Available at: <https://github.com/iabufarha/ArSarcasm>
- **ArSarcasm-v2 dataset**

- An extension of ArSarcasm dataset. It contains 15,548 tweets annotated for sarcasm, sentiment, and dialect.
 - The dataset was used to organise a shared task on Arabic sarcasm and sentiment analysis.
 - Available at: <https://github.com/iabufarha/ArSarcasm-v2>
- **Arabic iSarcasmEval dataset**
 - The first Arabic intended sarcasm dataset. It contains 5,503 sentences annotated for sarcasm and dialect.
 - The dataset was used to organise SemEval 2022 task 6 shared task, iSarcasmEval.
 - Available at: <https://github.com/iabufarha/iSarcasmEval>

Chapter 2

Background

This chapter includes background information that covers various topics. In this chapter, we provide an overview of Arabic, its variants and the challenges faced when working with it. We also provide an overview of sarcasm from a linguistic point of view, its features, and the factors that affect it. A general overview of Arabic NLP is also included along with an extensive survey of work on sarcasm detection in Arabic and English.

2.1 Arabic Language

Arabic is the most widely spoken Semitic language and is an official language in 22 countries with around 400 million native speakers (Darwish and Magdy, 2014). Furthermore, Arabic has a particular religious importance since it is the language of the Quran, the holy book of around 1.6 billion Muslims around the world.

There are three types of Arabic: Classical Arabic, Modern Standard Arabic (MSA) and Dialectal Arabic (DA). Classical Arabic resembles the language of the Quran, which is the old Arabic language, with many phrases that are not frequently used these days. MSA is the current unified form of Arabic which is taught in schools and used in media and news (Habash, 2010). DA is the colloquial language which is spoken in everyday life, this language differs from one country to another, and even varies inside the country itself. DA differs from MSA in many aspects as it sometimes does not follow a specific grammar and it has many words that are pronounced differently. It also contains many words that are either borrowed from other languages or specific to that dialect (Darwish and Magdy, 2014).

Arabic imposes the following challenges for NLP researchers who are willing to work on it:

- **Dialects variety:** As mentioned previously, Arabic has many different dialects that are considered regional and differ from one region to another. According to (Hamdi et al., 2016; Darwish and Magdy, 2014), Arabic dialects can be put into five major groups based on the region as follows:
 - Gulf: Saudi Arabia, United Arab Emirates, Oman, Kuwait, Qatar, Bahrain and Yemen.
 - Iraqi: Iraq.
 - Levantine: Palestine, Jordan, Syria and Lebanon.
 - Egyptian: Egypt and Sudan.
 - Maghribi: Morocco, Libya, Algeria, and Tunisia.

However, it is also worth mentioning that there are some slight differences between the dialects of countries within the same group (Alhumoud et al., 2015). For example, the word **زَاكِي** “zākī”¹ meaning “delicious” is used in Jordan and Palestine; whereas the word **طَيِّب** “tayyib” is used in Lebanon and Syria. Additionally, sometimes dialects might have different ways of expressing negation, which might be different from MSA or other dialects. For example, in the Palestinian dialect, a person would say **بِعْرِفْش** “b’rfsh” which means “I don’t know”, while a Lebanese or a Syrian would say **مَا بَعْرِفْ** “mā b’rf”. Negation in the Palestinian dialect is slightly similar to negation in old English as in saying (I know not). Generally, the above division is a geographic grouping; however, Arabic dialects are diverse and vary from one region to another. Figure 2.1² shows the distribution of Arabic dialects.

- **Morphological complexity:** Arabic is considered one of the morphologically complex languages. Given a single root, it is possible to derive and inflect many word forms with different meanings (Habash, 2010).
- **Ambiguity:** Compared to other languages, Arabic has an additional source of ambiguity coming from diacritics. The same word with different diacritics might have a completely different meaning. Since native speakers can easily infer diacritics from the context, people usually do not write them. An example of such ambiguity is the word **كَتَبَ** “kataba” which means “he wrote” and the word **كُتُبُ** “kutub” which means “books”. However, both words are written as **كتب**, without diacritics (Zayyan et al., 2016).

¹Arabic transliterations are using the Library of Congress standard (<https://www.loc.gov/catdir/cpsd/romanization/arabic.pdf>). The tool available on (<https://romanize-arabic.camel-lab.com/>) was used.

²Figure retrieved from https://en.wikipedia.org/wiki/Varieties_of_Arabic

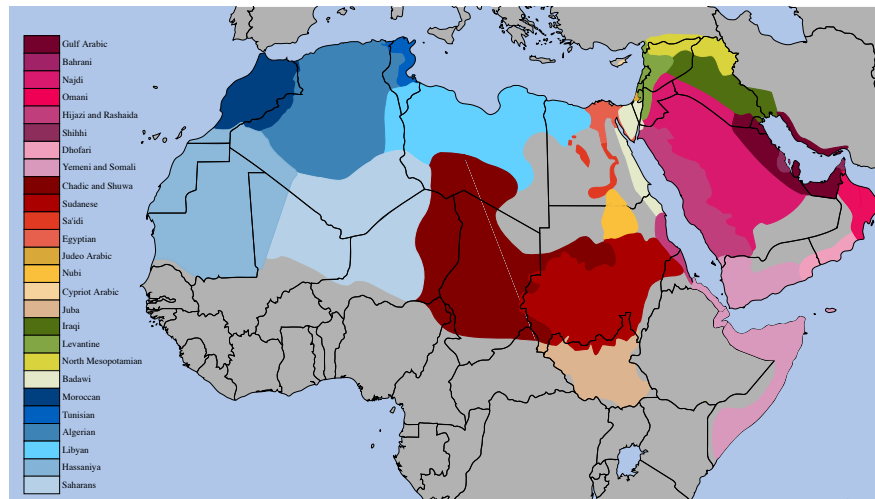


Figure 2.1: The geographic distribution of Arabic dialects.

- **Lack of resources:** one of the main challenges that researchers face when working on Arabic is the lack of resources such as corpora and lexicons. This becomes more prominent when dialects are to be considered because they differ from each other, which, in turn, requires specific resources for them in addition to MSA (El-Beltagy and Ali, 2013; Refaee and Rieser, 2014).

2.2 Sarcasm

2.2.1 Definitions

Sarcasm can be defined as a form of verbal irony that is intended to express contempt or ridicule (Joshi et al., 2017). Sarcasm is correlated with expressing the opinion indirectly, where the intended meaning is different from the literal one (Wilson, 2006). However, researchers and linguists cannot yet agree on a specific definition of what is considered to be sarcastic. According to Grice et al. (1975), sarcasm is a form of figurative language where the literal meaning of words is not intended, and the opposite interpretation of the utterance is the intended one.

Gibbs (1994) defines sarcasm as a bitter and caustic form of irony. According to Merriam Webster's dictionary³, sarcasm is “a sharp and often satirical or ironic utterance designed to cut or give pain”, while irony is defined as “the use of words to express something other than and especially the opposite of the literal meaning”. These

³<https://www.merriam-webster.com>

definitions are quite close to each other, yet each of them gives a different definition of sarcasm. While most of the literature assumes that sarcasm is a form of irony, [Justo et al. \(2014\)](#) argue that it is not necessarily ironic. Thus, sarcasm is always confused with other forms of figurative language such as metaphor, irony, humour and satire.

There have been many linguistic studies that described the aspects of sarcasm. [Campbell and Katz \(2012\)](#) studied the dimensions along which sarcasm might occur, which include failed expectation, pragmatic insincerity, negative tension, and the presence of a victim. According to [Wilson \(2006\)](#), sarcasm occurs whenever there is a disagreement between text and contextual information. For example, the sentence “*I love being ignored*” is considered sarcastic because of the disagreement between the contextual information that “being ignored” is an undesirable feeling and that the speaker claims to love it in the mentioned sentence. [Camp \(2012\)](#) identified four types of sarcasm: Propositional, where a sentence might appear as a proposition but has an implicit sentiment. Embedded, where the sentence would have an embedded contrast in sentiment. Like-prefixed, where a like-phrase would indicate an implied denial such as “like you care!”. Illocutionary, where non-textual clues would indicate the incongruity, such as rolling the eyes.

2.2.2 Socio-linguistic Variables

Since sarcasm is a very subjective form of expression, many variables can affect how people use or understand a sarcastic utterance. Such variables include:

- Gender: [Gibbs \(2000\)](#) noticed that men are more likely to speak sarcastically than women. [Rockwell and Theriot \(2001\)](#) also noticed similar behaviour. They also noticed that men are more likely to be sarcastic when interacting with men than when interacting with women. They also found that women are more likely to be sarcastic when interacting with men than with women.
- Age: [Phillips et al. \(2015\)](#) studied the correlation between age and the ability to interpret sarcasm expressed verbally and in video form. They found that older adults were poorer at understanding sarcastic intent compared with younger and middle-aged participants. A similar pattern was also noticed for the video tasks.
- Country/region: [Dress et al. \(2008\)](#) hypothesised that the regional language variation would result in variation in the use of verbal irony and sarcasm by Northern and Southern speakers in the United States. They found that Northern speakers are more likely to use sarcasm.

2.2.3 Intended vs Perceived Sarcasm

Most of the previous work on sarcasm detection utilised datasets that were annotated manually. The data would be collected and provided to third-party annotators to assign a label indicating whether a text is sarcastic. Dress et al. (2008) noticed that people from different regions defined sarcasm differently. Thus, and considering the factors mentioned in the previous section, the provided labels represent the annotators' perception of the given text, hence *perceived sarcasm*. *Intended sarcasm* represents the actual intentions of the author. Hence, the labels are provided by the text's author.

2.2.4 Arabic Sarcasm

In contrast to English, historically, Arabic linguists did not use the term *المفارقة* "Al-Mufāraqah" (irony). In the last century, Arabic linguists adopted translations of the definitions from English and inherited the same disagreement regarding how to define sarcasm and how it differs from irony (Andalib and Fare' Shirazi, 2019; Dawodi and Sharif, 2019)^{4[1,2]}. In (Andalib and Fare' Shirazi, 2019)^[1], the authors discuss the various definitions of sarcasm (*السخرية*, "Al-Sukhriyah") and contrast it to irony (*المفارقة*, "Al-Mufāraqah"). They show that the issue is the translations of the terms "irony" and "sarcasm" into Arabic and that irony (*المفارقة*) is based on contradiction and aims for mockery or sarcasm. Hence, irony and sarcasm are intertwined in Arabic and cannot be separated.

Arabic literature is rich with other linguistic phenomena that are similar to irony. Those include: (1) *التورية* "Al-Tawriyah" (double entendre), where a word/phrase is used and has a double meaning, and the speaker intends one of them; (2) *التعريض* "Al-Ta'rīd", where a speaker implies things without mentioning them explicitly; (3) *المدح بما يشبه الذم* (praise using slander); (4) *الذم بما يشبه المدح* (slander using praise); (5) *الاستعارة* (metaphor); (6) *الكناية* (metonymy), where something is referenced implicitly using its features or descriptions; and others. The shared theme among these phenomena and irony is that they rely on the duality of meaning (Dawodi and Sharif, 2019)^[2].

Apart from the terminology, sarcasm has been present in the Arabic language literature. Sarcasm has been used as an implicit tool to express **humour**, **mockery**, **insult**, **slander**, or **criticism** of rulers or current events (Al-Mansoori, 2012)^[3]. One of the most famous sarcastic works in Arabic prose literature is *البخلاء* (The Book of Mis-

⁴Numbers in the brackets are used for referencing the Arabic bibliography.

ers) by الجاحظ “Al-Jāhiz”, which was written in the 9th century (Ibrahim, 1987)^[4]. In his book, Al-Jāhiz describes and tells stories about misers from his town, describing their behaviour and inner thoughts.

Sarcasm was also present in Arabic poetry. An example of that is the rivalry between the Umayyad⁵ poets جرير “Jarīr” and الفرزدق “Al-Farazdaq”, which was known as النقااض “Al-Naqā'id” (antitheses) (Al-Aboudi and Khalaf, 2014)^[5]. In the following example, Jarīr implies that Al-Farazdaq is weak and that he will not be fulfilling his threats to Marba’:

زَعَمَ الْفَرَزْدَقُ أَنْ سَيَقْتُلُ مَرْبَعًا أَبْشِرْ بِطَوْلِ سَلَامَةٍ يَا مَرْبَعُ

(Al-Farazdaq claimed he will kill Marba’Oh Marba’, then expect a lengthy life)

Sarcasm is also present in more modern poems. In the following example, Mahmoud Darwish, who was a prominent Palestinian poet and writer and widely regarded as one of the most important literary figures in the Arab world, in his poem تحد (Defiance), says the following:

شدوا وثاقي (tighten my cuffs)
 وامنعوا عني الدفاتر (take away my notebooks)
 والسجائر (and ciggarettes)
 وضعوا التراب على فمي (and put dirt on my mouth)
 فالشعر دم القلب (but poetry is the blood for the heart)
 ملح الخبز (the salt for the bread)
 ماء العين (the tears for the eye)
 يكتب بالأظافر والمحاجر (it is written with nails and stones)
 والحناجر (and throats)

In those verses, the poet asks the Israeli occupation to silence him, but he actually says that as a sign of defiance and even if they did that, they would not stop him from defying them (Ragab, 2009)^[6].

In modern days, sarcasm is still present and highly used among Arabic speakers. A major use of sarcasm these days is to criticise authorities and comment on ongoing events in a sarcastic/satirical way. For example, Al-Hudood⁶ is a website that provides news written sarcastically. Arabic TV channels have many sarcastic shows that

⁵Refers to the Umayyad Caliphate period 661–750 CE.

⁶<https://alhudood.net/>

sarcastically comment on cultural or political issues, such as ⁷فوق السلطة (Beyond Authority) and *وطن ع وتر* (A Country on A String).

Regarding social media, people also use sarcasm in the content they post and the discussions they participate in. Sarcastic content on social media can be textual or multimodal in the form of memes. One of the most common ways to create memes is to utilise scenes from famous Egyptian movies. Those are usually comedy movies that are famous in Egypt and other Arab countries. The memes are used with either the original quote from the scene or with a new quote resembling a similar situation to the original one. Figure 2.2 shows examples of memes and quotes. Figure 2.2a shows a meme that is used when the speaker is not satisfied with what someone said or thinks that it is not useful. Figure 2.2b is used to indicate the fear of the future based on current events or news.



(a) The person says “Thanks.. Go back to your mother”, which is used to ask the other person to stop talking.



(b) The person says “Hey guys, we will see black days!”, which means that we are heading to bad times.

Figure 2.2: Snippets from movies containing sentences that are used as sarcastic responses on Arabic social media.

Sarcastic content on social media is usually dialectal and contains references to real-life events or people. Table 2.1 shows some sarcastic Arabic sentences. In the first example, the speaker tells someone that she will not become Justin Beiber, who is a famous singer. This reference is used to refer to either the singer being famous or having a good voice and being able to sing. The use of this reference in such a context is an implied mocking of the other person’s voice or desire to be famous. The second example references an augmented reality mobile game named Pokemon Go, where you must search for Pokemons, which are animal-like characters from Anime.

⁷<https://www.aljazeera.net/program/beyond-authority/>

In the sentence, the speaker mocks the game and implies that he/she will not play it because he/she is so lazy to the degree that he/she will not even move to turn the lights off in the room.

<i>ID</i>	<i>Sentence</i>	<i>Dialect</i>
1	<p>لن تصبجي جستن بيبير</p> <p>LT: You won't become Justin Bieber.</p> <p>IM: Your voice is bad or you won't become famous</p>	MSA
2	<p>بو كيمون ايه اللي اقوم اضور عليه .. ده انا بكسل اقوم اطفي نور الاوضه !!</p> <p>LT: What Pokemon that I would look for! I am too lazy to turn off my room's light.</p> <p>IM: I don't want to move.</p>	Egypt

Table 2.1: Examples of some sarcastic sentences in different dialects. LT stands for literal translation. IM stands for intended meaning.

2.3 Arabic Natural Language Processing

Over the last two decades, Arabic natural language processing (NLP) started to receive attention from the research community. Work on Arabic NLP focused on two directions: (1) creating resources such as datasets and corpora; (2) targeting NLP tasks such as morphological analysis. Early work on Arabic NLP started in the 1980s and focused on modern standard Arabic (MSA). The first Arabic morphological analyser was built by Sakhr Software. Sakhr followed that by developing the first Arabic semantic and syntactic analyser, English to Arabic machine translation, and Arabic text-to-speech. In early 2000s, Arabic NLP received more attention, especially in the US due to the events around that time. Work in this period included machine translation ([Habash and Sadat, 2006](#)), text summarisation ([Douzidia and Lapalme, 2004](#)), and part-of-speech (POS) tagging ([Beesley, 1998](#); [Habash and Rambow, 2005](#)).

In the 2010s, Arabic NLP research moved to the Arab world. This was also accompanied with the widespread of social media platforms which led to an abundance of user-generated content. However, the content on social media platforms is usually in dialectal Arabic (DA) and can be mixed with MSA. Thus, Arabic NLP researchers started focusing on DA and the other tasks that can help understand this content, such as dialect identification ([Zaidan and Callison-Burch, 2014](#); [Elfardy et al.,](#)

2014; Bouamor et al., 2019; Abdul-Mageed et al., 2020, 2021b), sentiment analysis (Mourad and Darwish, 2013; Abdul-Mageed et al., 2014), emotion detection (Alhuzali et al., 2018), and offensive language detection (Mubarak et al., 2020). Darwish et al. (2021) provide a general overview of Arabic NLP.

2.3.1 Arabic Subjectivity and Sentiment Analysis

Early works on opinion mining of Arabic content focused on analysing subjectivity and sentiment. Subjective content refers to content expressing opinions, feelings, or speculations. Arabic NLP researchers focused on analysing sentiment in subjective content, i.e. sentiment analysis. Sentiment analysis (SA) is the process of analysing and identifying the polarity/sentiment expressed in a text.

Early works on Arabic SA focused on Modern Standard Arabic such (Abbasi et al., 2008; Abdul-Mageed et al., 2011). In (Abdul-Mageed and Diab, 2012), the authors proposed a corpus for subjectivity and sentiment analysis. Later, the focus shifted to analysing social media content and analysing dialectal Arabic (DA). Mourad and Darwish (2013) introduced a dataset of tweets for subjectivity and sentiment analysis along with baselines. Abdul-Mageed et al. (2014) proposed an SA system for social media. In their work, they experimented with a large set of features. Moreover, In (Abdul-Mageed, 2017, 2019), the authors studied different possible ways of handling the morphological richness of Arabic for the task of SA.

In (Abdulla et al., 2013), the authors collected their own dataset of 2000 tweets. They experimented with different sentiment analysis approaches; their best model was an SVM which achieved an accuracy of 87%. Soliman et al. (2014) targeted social media where they tried to handle the dialectal variation through building their own lexicon, namely slang sentimental words and idioms lexicon (SSWIL). They utilised the lexicon and an SVM classifier, which achieved an accuracy of 87%. In (Al-Ayyoub et al., 2015), the authors built a lexicon-based sentiment analysis system that utilises their own lexicon. The model was tested on manually collected and labelled tweets; they achieved an accuracy of 87%.

Dahou et al. (2016) proposed a set of Arabic word embeddings to be used for Arabic sentiment analysis. In order to build the embeddings, they used a corpus of around 3.4 billion words. A CNN-based model, which utilises the newly created embeddings, was used to perform sentiment analysis on LABR book reviews dataset (Aly and Atiya, 2013), Arabic Sentiment Tweets Dataset (ASTD) (Nabil et al., 2015) and

other datasets. Another word embeddings set was proposed by [Altowayan and Tao \(2016\)](#). They used the embeddings as features to be fed to the classifier. In their experiments, SVM was the best classifier.

[Alayba et al. \(2017\)](#) proposed a new dataset for opinions on health services, which was collected from Twitter. They experimented with different sentiment analysis approaches on the new dataset, their experiments included SVM, Naive Bayes and CNNs. The best classifier was SVM with an accuracy of 91%. [Al Sallab et al. \(2015\)](#) experimented with various deep learning models such as recursive auto-encoder (RAE), deep belief networks (DBN) and deep auto-encoder (DAE). In their work, they relied on the bag of words (BoW) representation of text and some lexical features.

In SemEval 2017, Arabic was added to one of the sentiment analysis tasks ([Rosenthal et al., 2017](#)). The winner was NileTMRG team ([El-Beltagy et al., 2017](#)), where they used a large set of hand-engineered features that covers a large variety of syntactic, lexical and statistical features. They used a complement Naive Bayes classifier which achieved an average recall of 0.583 and F^{PN} score⁸ of 0.61. The runner-up was SiTAKA team ([Jabreel and Moreno, 2017](#)), they used a combination of features such as bag-of-words and lexical features. Moreover, they introduced some features that are based on the word embedding vectors such as sum, min, max and standard deviation. The classifier of choice was SVM which achieved an average recall of 0.55 and F^{PN} score of 0.571.

Additionally, [Alayba et al. \(2018\)](#) experimented with deep learning models for Arabic sentiment analysis. In their work, they built a model that is based on a combination of CNN and LSTM. They tested their model on different datasets such as Twitter dataset (Ar-Twitter) and Arabic Health services dataset, which they introduced in a previous work. The final model achieved an accuracy of 88.1% and 94.3% on the datasets, respectively.

[Al-Smadi et al. \(2018\)](#) compared the performance of SVM against an RNN-based model in building an aspect-based⁹ sentiment analysis system. They tested the model on a dataset for Arabic hotels' reviews, which was part of SemEval 2016 ([Pontiki et al., 2016](#)). In their approach, they used a combination of lexical, syntactic, semantic and morphological features. Their results showed that SVM, which achieved an accuracy of 95%, was better than the RNN model, which achieved an accuracy of 87%, for that

⁸ F^{PN} is the macro average F-score for the positive and negative classes only.

⁹Aspect-based sentiment analysis works through identifying the aspects/features of a product/service and then finding the sentiment related to each of them.

specific task.

Furthermore, [Al-Smadi et al. \(2019\)](#) proposed an aspect-based sentiment analysis system, where they created a model based on a character-level BiLSTM combined with conditional random field (CRF) that was responsible for extracting the aspect opinion target expression. For the sentiment classification, they used an LSTM based model. They tested their models on the Arabic hotels' reviews dataset where they had an improvement of around 39% with an F-score of roughly 70%.

[Al-Twairesh and Al-Negheimish \(2019\)](#) proposed to learn sentiment-specific word embeddings. They used the new embeddings to test and compare their effectiveness against generic embeddings. In their experiments, they tested different models including deep learning. They found that generic embeddings outperform sentiment-specific ones. In [\(Oussous et al., 2020\)](#), the authors experimented with different deep learning approaches on a corpus that they manually collected from multiple resources such as Twitter, YouTube, and Facebook. They also propose a framework that provides text preprocessing and sentiment classification capabilities.

Table 2.2 provides a summary of the previous approaches in Arabic SA. There have been many approaches introduced for Arabic SA. However, as could be noticed, each of these approaches is tested on a different dataset. This makes it difficult to identify the best approach among them. In our study [\(Abu Farha and Magdy, 2021\)](#), we compare most of the approaches discussed in the literature on standardised benchmark datasets to have a deep comparative analysis of the effectiveness of these approaches on multiple Arabic datasets.

Table 2.2: Summary of previous approaches in Arabic sentiment Analysis. The polarities are positive (POS), negative (NEG), or neutral (NEU).

Article	Dataset/source	Features	Approach	Polarity
Abdul-Mageed et al. (2011)	Penn Arabic Treebank	Domain, unique words, n-grams, lexicon-based features	SVM	POS/NEG
Abdul-Mageed et al. (2012)	Tweets, Wikipedia, Forums	Domain, unique words, n-grams, POS tags, lexicon-based features	SVM	POS/NEG
Abdulla et al. (2013)	Tweets	N-grams	SVM, NB, KNN, D-tree, unsupervised (lexicon-based)	POS/NEG
Mourad and Darwish (2013)	Tweets	POS tags, lexicon-based features	Naive Bayes	Subjectivity, POS/NEG
Soliman et al. (2014)	Comments from Facebook and news websites	N-grams, lexicon-based features	SVM	POS/NEG
Al-Ayyoub et al. (2015)	Tweets	-	Lexicon-based	POS/NEG/NEU
Al Sallab et al. (2015)	Arabic Treebank	N-grams, lexicon-based features, word embeddings	Neural networks, deep auto-encoder, deep belief network	POS/NEG
Dahou et al. (2016)	LABR, ASTD, Arabic Gold-Standard Twitter Sentiment Corpus	Word embeddings	CNN	POS/NEG
Altowayan and Tao (2016)	LABR, MPQA, ASTD, ArTwitter	Word embeddings	SVM	POS/NEG
Alayba et al. (2017)	Tweets about health service	N-grams	SVM, NB, Logistic regression	POS/NEG
El-Beltagy et al. (2017)	SemEval-2017	N-grams, lexical features, lexicon-based features	Complement NB	POS/NEG/NEU
Jabreel and Moreno (2017)	SemEval-2017	Word embeddings, syntactic features, n-grams, lexicon-based features	SVM	POS/NEG/NEU
Al-Smadi et al. (2018)	Arabic hotels' reviews (SemEval-ABSA16)	POS tags, NER feature, morphological features, n-grams, lemmas, stems, word embeddings	SVM, RNN	POS/NEG/NEU
Alayba et al. (2018)	Arabic Health Services dataset, ArTwitter, ASTD	Word embeddings, character embeddings	CNN-LSTM	POS/NEG
Al-Smadi et al. (2019)	Arabic hotels' reviews (SemEval-ABSA16)	Word embeddings	LSTM	POS/NEG/NEU
Abdul-Mageed (2019)	Arabic Treebank	Domain, unique words, n-grams, POS tags, lexicon-based features	SVM	POS/NEG

2.3.2 Arabic Language Models

The introduction of Bidirectional Encoder Representation from Transformers (BERT) (Devlin et al., 2019) led to a revolution in the NLP world. Since then, many other models have been released, such as ELECTRA (Clark et al., 2020), GPT-1/2/3 (Radford et al., 2019; Brown et al., 2020) and RoBERTa (Liu et al., 2019). Those models helped achieve state-of-the-art results on different tasks such as sentiment analysis, named entity recognition (NER), sentence completion and others. However, those models were trained mostly on English data, while others included data from other languages such as the multilingual BERT (Devlin et al., 2019). Recently, Arabic NLP researchers started training Arabic variants of these models such as the works of (Antoun et al., 2020, 2021a; Chowdhury et al., 2020; Abdul-Mageed et al., 2021a). AraBERT (v0.1/v1) (Antoun et al., 2020) was built using the same architecture as BERT-base (Devlin et al., 2019). AraBERT was trained using a combination of different Arabic news corpora. The authors utilised Farasa (Abdelali et al., 2016) for the preprocessing and segmentation, then they trained a SentencePiece tokenizer (Kudo, 2018) on the segmented text with a vocabulary of 60K subword tokens. The authors also released AraBERT (v0.2/v2), which was trained on a larger dataset of 77GB of text. The authors trained two variants of AraBERT based on BERT-base and BERT-large architectures. AraBERT was evaluated on three tasks: named entity recognition (NER), question answering and sentiment analysis. In another work, the authors released AraELECTRA (Antoun et al., 2021a), which is trained using the same architecture and procedure used to train the original ELECTRA model (Clark et al., 2020). AraELECTRA was trained using the same preprocessing used with AraBERT(v2) and using the same data. AraELECTRA was tested only for question answering task, where it achieved state-of-the-art results on multiple datasets. In (Antoun et al., 2021b), the authors released AraGPT2, which is based on the original GPT2 (Radford et al., 2019) architecture, this model was also trained on the same data used for AraELECTRA and AraBERT(v2). Since GPT2 is trained using causal language modelling objective, the authors did not test the model on any datasets and relied on the perplexity reported during training.

In (Lan et al., 2020a), the authors pre-trained a customised bilingual BERT, GigaBERT, that is designed specifically for Arabic NLP and English-to-Arabic zero-shot transfer learning. The training data was around 13M news articles collected from different sources. They also augmented their data with code-switched samples to improve

the cross-lingual performance. GigaBERT was evaluated on multiple NLP tasks such as: NER, part of speech (POS) tagging, relation extraction and argument role labelling. In (Abdul-Mageed et al., 2021a), the authors proposed two new Arabic-specific BERT models, ARBERT and MARBERT. For ARBERT, they used BERT-base architecture and 61GB of text as training data. The training data for ARBERT is mostly in modern standard Arabic (MSA) with a small portion in Egyptian dialect. For MARBERT, the authors aimed to improve the model's ability to handle dialectal Arabic. They used 1B tweets (128GB of text) as training data. Abdul-Mageed et al. (2021a) provide an extensive evaluation of their models on many tasks such as sentiment analysis, dialect identification, NER, and others. In (Chowdhury et al., 2020), the authors introduced a new Arabic BERT (QARiB). In their work, the authors tried to improve the performance of the model through diversifying the training data. In their experiments, they show that a BERT model trained on a mixture of formal and informal data has much better generalization power compared to BERT models that are trained on formal text only. QARiB was evaluated only on a text categorization task.

2.4 Computational Sarcasm Analysis

2.4.1 English Sarcasm Detection

The literature has a large amount of work on sarcasm and irony detection, which vary from collecting datasets to building detection systems.

One of the early works on English sarcasm/irony detection is the work of Davidov et al. (2010), where the authors created a dataset from Twitter using specific hashtags such as #sarcasm and #not, which indicate sarcasm. This way of data collection is called distant supervision, where data is collected based on some specific content that it bears. Distant supervision is the most common approach to collect sarcastic content from Twitter, where the hashtag #sarcasm and others are used. Some other works that utilised distant supervision to create Twitter datasets include (Ptáček et al., 2014; Bamman and Smith, 2015; Bouazizi and Ohtsuki, 2016). Davidov et al. (2010) mention that the use of the #sarcasm hashtag is possible but not reliable, and they used it as a search anchor. Also, such hashtags can be useful in cases of subtle sarcasm which might not be easily understood.

Ptáček et al. (2014) targeted both English and Czech in their work. They collected data for both languages using distant supervision through #sarcasm and its equivalent

in Czech. They utilised various sets of features that include n-grams and character n-grams, POS-tags-related features, word patterns, emoticons, word case, punctuation, etc. They experimented with multiple classifiers such as SVM and Maximum Entropy (MaxEnt). On Czech, SVM combined with character n-grams and pattern features achieved a macro-F1 of 0.582. MaxEnt was superior for English achieving a macro-F1 of 0.94.

[Justo et al. \(2014\)](#) worked on sarcasm and nastiness detection. They utilised a set of statistical cues that were extracted from the data, along with linguistic features such as POS tags n-grams. They also included LIWC features, length and sentiment. For the experiments, they utilised the dataset in ([Lukin and Walker, 2013](#)), where they tested the effectiveness of various combinations of the features. They experimented with Naive Bayes and a rule-based classifier. The experiments showed that sarcasm is more subtle and harder to detect than nastiness.

[Bamman and Smith \(2015\)](#) collected data using a similar approach, where they utilised #sarcasm and #sarcastic hashtags. In their work, they tried to incorporate contextual information in addition to the textual and lexical features. They utilised tweet features such as n-grams, part of speech (POS) tags, tweet and word sentiment, and other stylistic features. They also utilised some author-related features such as the most frequent topics and terms in the author's history, the author's historical sentiment and other profile information. They also included audience features such as the historical topics of interaction with the author and the historical communication between the author and the audience. Finally, they also utilised environment-related features such as the unigram features of the tweet and the pairwise Brown features between the tweet and the responses. In their work, they used a binary logistic regression which achieved an accuracy of 79% when used with all the mentioned features.

[Rajadesingan et al. \(2015\)](#) followed a behavioural approach to detect sarcasm. They introduced a behavioural framework to represent sarcasm. The framework identifies a set of characteristics for sarcasm generation. In their work, they provide a set of features for each of these characteristics. Those include contrast in sentiment, readability features, emotions and language features such as grammar skills and structural variation. They tested their framework on a set of tweets, which were collected using distant supervision. They experimented with a J48 decision tree, L1-regularised logistic regression, and L1-regularised L2-loss SVM, achieving accuracies of 78.06%, 83.46%, and 83.05% respectively.

[Joshi et al. \(2015\)](#) tried to detect sarcasm by utilising context incongruity. They

considered two cases for identifying sarcasm: The first is explicit incongruity, where the positive and negative sentiment would appear in the same sentence. The second is implicit incongruity, which is expressed through phrases of implied sentiment. They used the following to represent explicit incongruity: number of sentiment incongruities, largest positive/negative subsequence, number of positive/negative words and lexical polarity. For implicit incongruity, they used a set of sentiment-bearing verbs and phrases. They also used n-grams features, capitalisation and punctuation. They used an SVM classifier on a set of tweets they collected using distant supervision, which achieved an F1-score of 0.64.

[Khattri et al. \(2015\)](#) proposed a model for sarcasm detection that is based on the contrast in sentiment, where a piece of text is considered to be sarcastic if the author expresses opposing sentiments about the same topic. The model utilises the sentiment expressed in a given tweet and the author's historical sentiment towards the topic in the target tweet. In their work, they also consider the contrast in sentiment within the target tweet itself and they propose various methods to combine the results of both approaches. They tested their model on a set of tweets collected using distant supervision, where they achieved an F1-score of 0.88.

[Bouazizi and Ohtsuki \(2016\)](#) proposed a pattern-based approach to detect sarcasm. Their approach is based on four sets of features: First, sentiment-related features such as the counts of positive words, negative words and sarcastic emoticons. Second, punctuation-related features such as the number of exclamation marks, question marks and dots. Third, syntactic and semantic features, such as the use of uncommon words, the number of uncommon words, number of laughing expressions. Fourth, pattern-related features where they look for a set of patterns which they identified. They experimented with multiple classifiers such as SVM, Random Forest, Maximum Entropy and KNN. Random Forest was the best model, which achieved an accuracy of 83.1%.

[Zhang et al. \(2016\)](#) used neural networks to detect sarcasm. Their model is based on bidirectional gated recurrent unit (Bi-GRU). They tested their model on Twitter data provided by [Rajadesingan et al. \(2015\)](#). Their approach showed a significant improvement over the approach provided in the original paper. The model achieved an F1-score of 0.79 when using only the tweets and an F1-score of 0.91 when combined with context tweets. [Ghosh and Veale \(2016\)](#) also utilised neural networks in their work, where they experimented with recursive SVM and a combination of convolutional neural network (CNN) and long short-term memory (LSTM). They tested their models on a dataset of tweets collected using distant supervision. Their best model

was a CNN-LSTM model which achieved an F1-score of 0.92 on their data and an F1-score of 0.88 on the data from (Riloff et al., 2013). Amir et al. (2016) proposed a model that incorporates context into the detection process, which is done through the utilisation of user embeddings. Their model is based on a CNN that takes both the textual information and the user embedding as input. They tested their model on the data provided by Bamman and Smith (2015), where they achieved an accuracy of 87.2% compared to 85.1% reported in the original paper.

Joshi et al. (2018) utilised sentence completion to detect sarcasm. The approach is based on detecting the incongruity between the observed and the expected word. They experimented with two approaches: The first is considering all words. The second approach only considers incongruous words. In both approaches, the similarity score is computed between the observed word and the expected word, if it is below a specific threshold then the sentence is considered sarcastic. They used Context2vec (Melamud et al., 2016) to perform the sentence completion. They tested their approach on the datasets from (Riloff et al., 2013) and (Walker et al., 2012) achieving an F1-score of 0.80 and 0.45 respectively.

Tay et al. (2018) propose Multi-dimensional Intra-Attention Recurrent Network (MIARN) and Single-dimensional Intra-Attention Recurrent Network (SIARN) models, which are attention-based models for sarcasm detection. The approach they followed is based on the concept of contrast or incongruity, where the model utilises the attention mechanism to look for incongruity between the words. They tested their model on various datasets, including (Ptáček et al., 2014) and (Riloff et al., 2013). MIARN model achieved F1-scores of 0.86 and 0.70, while SAIRN achieved F1-scores of 0.84 and 0.73.

Hazarika et al. (2018) proposed a contextual sarcasm detection model. The model takes into consideration both content and context of a relevant piece of text. In their work, they used a CNN to extract content-based features. Regarding contextual features, they utilised user embeddings and discourse features. To create user embeddings, the authors utilised a set of stylometric and personality features. They tested their model on a set of sarcastic Reddit posts (Khodak et al., 2018), where they achieved an F1-score of 0.77.

Oprea and Magdy (2019) explored the effect of contextual information to detect intended and perceived sarcasm. In their work, they suggest a procedure to improve the user embeddings in (Hazarika et al., 2018) by giving a higher weight to more recent tweets. They experimented on the datasets provided by Ptáček et al. (2014) and Riloff

et al. (2013). In another work, Oprea and Magdy (2020a) proposed a new dataset that captures intended sarcasm. They collected their data using an online survey, where they asked the participants to provide sarcastic and non-sarcastic tweets. They also asked them to provide an explanation for the sarcastic text and how would they convey the same idea in a direct way. In their work, they experimented with various models such as attention-LSTM, CNNs and SIARN/MIARN (Tay et al., 2018). They found that the available models perform poorly on intended sarcasm. Baruah et al. (2020) experimented with BERT (Devlin et al., 2019) for sarcasm detection. They also experimented with other models such as Bi-LSTM and SVM. They tested these models on two separate datasets, one from Twitter and the other is from Reddit. BERT was the best model, where it achieved F1-scores of 0.74 and 0.66 on the Twitter and Reddit datasets respectively.

2.4.2 Arabic Sarcasm Detection

Work on Arabic sarcasm is scarce and limited to a few attempts. The earliest work on Arabic sarcasm/irony is (Karoui et al., 2017), where the authors created a corpus of Arabic tweets, which they collected using a set of political keywords. They filtered sarcastic content using distant supervision, where they used the Arabic equivalent of #sarcasm such as #سخرية, #مسخرة, #تهكم and #استهزاء. The result was a set of 5,479 tweets distributed as follows: 1,733 ironic tweets and 3,746 non-ironic. In their work, the authors utilised various features to experiment with their data. These features include punctuation marks, emoticons, quotations, opposition words, sentiment features, shifters features and contextual clues. They experimented with various classifiers such as SVM, Naive Bayes, Logistic Regression, Linear Regression. Random Forest was the best model, where it achieved an F1-score of 0.73.

Ghanem et al. (2019) organised a shared task competition for Arabic irony detection. They collected their data using distant supervision and used similar Arabic hashtags to the ones in (Karoui et al., 2017). In addition, they manually annotated a subset of tweets, which were sampled from ironic and non-ironic sets. The first place were Khalifa and Hussein (2019), who achieved an F1-score of 0.85. In their work, they utilised a set of features that include word n-grams, topic modelling features, sentiment features, statistical features and word embeddings. They experimented with multiple classifiers such as Bi-LSTM, Random Forest, XGBoost. Zhang and Abdul-Mageed (2019) were ranked second, where they achieved an F1-score of 0.82. In their

work, they experimented with Bi-GRUs and utilised the multilingual BERT (Devlin et al., 2019) to perform the task.

Ghanem et al. (2020) explored sarcasm in a multilingual context. In their work, they targeted English, French and Arabic, where they used the datasets provided by Karoui et al. (2017) (Arabic), Ptáček et al. (2014) (English) and Benamara et al. (2017) (French). For the Arabic dataset (Karoui et al., 2017), they added more tweets using distant supervision. In their work, they explored both monolingual and multilingual sarcasm detection. For the monolingual detection, they utilised a set of features that include punctuation, negative/positive emotions, quotations, and other language dependent features such as negation. Those features were used to train a Random Forest classifier. They also used a CNN based model with word embeddings as an input. Regarding the multilingual models, they used similar features and models.

2.4.3 Sarcasm and Irony Detection in Other Languages

Most of the work on sarcasm detection has been focused on English. In this section, we go over other attempts to tackle sarcasm detection in other languages, these include Italian, Greek and others.

Barbieri et al. (2014a) proposed a system to detect sarcasm and irony in Italian texts. They collected their data from specific Twitter accounts along with some satirical blogs. In their work, they trained a Decision Tree classifier on various sets of features. Those include bag-of-words, sentiment, POS tags, word frequencies and others. del Pilar Salas-Zárate et al. (2017) tried to detect satire in Spanish texts, they targeted Spanish in Spain and Mexico. In their work, they utilised a set of psycholinguistic features that were extracted using LIWC. They tested multiple classifiers such as Bayesian Networks, SVM and decision trees. Liu et al. (2014) proposed a feature-based model to detect Chinese sarcasm. Their features included punctuation, recurring symbols, words' sentiment and some patterns. In their work, they used an ensemble of classifiers including SVM, Naive Bayes and Maximum Entropy.

Bharti et al. (2017) proposed a framework to detect Hindi sarcasm. In their work, they utilised news headlines as the context to identify sarcasm within tweets. They also utilise key-phrase extraction to retrieve the set of news related to the tweet. Then, they utilise sentiment information to identify the presence of sarcasm in a tweet, where a tweet is sarcastic if there is a contradiction between its sentiment and the sentiment of the related news. Charalampakis et al. (2016) compared the performance of semi-

supervised approaches to supervised ones for detecting irony in Greek political tweets. In their work, they utilised different kinds of tweets. Those include the use of some special character, the degree of rarity of a word in the data, WordNet features, emoticons and lexical features. For the classification part, they used collective-tree semi-supervised classifier. For the supervised ones, they used SVM, J48, Naive Bayes and Random Forest. Table 5.1 summarises the related works on sarcasm detection.

2.5 Research Gap

Given the survey in the previous section, it is clear that most of the work on sarcasm detection has been focused on English. The work on Arabic has been limited to a few attempts, indicating a gap that needs to be filled. More resources and datasets should be created for Arabic sarcasm. Also, more work targeting building detection models is needed. In this thesis, we try to fill this gap by creating and sharing new datasets, experiments, and analyses.

Table 2.3: Summary of sarcasm detection approaches in the literature.

Article	Language	Data	Features	Classifier
Joshi et al. (2018)	English	Twitter Riloff et al. (2013), Walker et al. (2012)	context incongruity using a sentence completion system Melamud et al. (2016)	-
Práček et al. (2014)	English, Czech	Twitter (#sarcasm)	n-grams, POS tags, word patterns, emoticons, punctuation	SVM, Maximum Entropy
Justo et al. (2014)	English	Lukin and Walker (2013)	Statistical cues, POS tags, LIWC, sentiment	Naive Bayes, rule-based classifier
Banman and Smith (2015)	English	Twitter(#sarcasm, #sarcastic)	Tweet features: n-grams, POS-tags, sentiment Author features: historical topics and sentiment, profile information Audience information: historical interactions, historical topics environment features	Binary logistic regression
Rajadesingan et al. (2015)	English	Twitter	contrast in sentiment, readability features, emoticons, language features such as grammar skills and structural variation	J48 Decision Tree
Joshi et al. (2015)	English	Twitter	number of sentiment incongruities, largest positive/negative subsequence, number of positive/negative words and lexical polarity	SVM
Khattri et al. (2015)	English	Twitter	Sentiment contrast within a tweet or with historical tweets	-
Bouazizi and Ohtsuki (2016)	English	Twitter (#sarcasm)	sentiment features: counts of negative/positive words, sarcastic emoticons. Punctuation features: number of exclamation marks, number of question marks, number of dots, etc. Syntactic and semantic features. pattern-related features	SVM, KNN, Random forest, Maximum Entropy
Zhang et al. (2016)	English	Rajadesingan et al. (2015)	word embeddings	Bi-GRU
Choshi and Veale (2016)	English	Twitter	word embeddings, constituency parsing trees	CNN-LSTM, recursive SVM
Amir et al. (2016)	English	Banman and Smith (2015)	word embeddings, user embeddings	CNN
Tay et al. (2018)	English	Twitter (Práček et al., 2014), Reddit (Khodak et al., 2018)	word embeddings	Multi/single dimensional intra-attention + LSTM
Hazarika et al. (2018)	English	Reddit	word embeddings	CNN
Oprea and Magdy (2019)	English	Twitter (Práček et al., 2014), (Riloff et al., 2013)	user embeddings, discourse features, stylistometric and personality features	Tay et al. (2018); ?
Oprea and Magdy (2020a)	English	Twitter (intended)	(Tay et al., 2018; Hazarika et al., 2018) word embeddings	Att-LSTM, CNN, Tay et al. (2018)
Barua et al. (2020)	English	Twitter (Práček et al., 2014), Reddit (Khodak et al., 2018)	word embeddings, contextual word embeddings, n-grams	SVM, Bi-LSTM, BERT
Barbieri et al. (2014a)	Italian	Twitter	bag-of-words, sentiment, POS tags, word frequencies	Decision Tree
Liu et al. (2014)	Chinese	Sina Weibo	punctuation, recurring symbols, words' sentiment	SVM, Naive Bayes, Maximum Entropy
Charalampakis et al. (2016)	Greek	Twitter	special character, rarity of a word, WordNet features, emoticons and lexical features	Collective-Tree, SVM, J48, Naive Bayes, Random Forest
del Pilar Salas-Zarate et al. (2017)	Spanish	Twitter	LIWC	Bayesian Networks, SVM, Decision Tree
Bharti et al. (2017)	Hindi	Twitter	sentiment	Rule-based classifier
Karoui et al. (2017)	Arabic	Twitter	punctuation marks, emoticons, quotations, opposition words, sentiment features, shifters features, contextual clues	SVM, Naive Bayes, Logistic Regression, Linear Regression
Khalifa and Hussein (2019)	Arabic	Twitter	word n-grams, topic modelling features, sentiment features, statistical features, word embeddings	Bi-LSTM, Random Forest, XGBoost
Zhang and Abdul-Mageed (2019)	Arabic	Twitter	word embeddings, contextual word embeddings	Bi-GRU, BERT
Ghanem et al. (2020)	Arabic, English, French	Twitter	word embeddings, punctuation, negative/positive emoticons, quotations	CNN, Random Forest

Chapter 3

Annotating Arabic Sentiment Datasets for Sarcasm

In Chapter 2, we provided a literature survey that shows the scarcity of works on Arabic sarcasm detection. This chapter provides an overview of our effort to create an Arabic sarcasm dataset and answer the following research questions:

- **RQ1:** Is sarcasm used across Arabic social media? If so, how prominent is it?
- **RQ2:** How does sarcasm affect the performance of sentiment analysers?
- **RQ3:** How to collect and reliably annotate a dataset for Arabic sarcasm detection?

To create an Arabic sarcasm dataset and answer these questions, we re-annotated available Arabic sentiment datasets for sarcasm. The resulting data show that sarcasm is quite present in Arabic sentiment datasets and that it can be disruptive for sentiment analysers. The analysis provided in this chapter discusses some of the issues that might arise when annotating subjective content, which would affect the reliability of the resulting datasets. In addition, we provide the details of the shared task on Arabic sarcasm, which utilises an extended version of the proposed dataset. This chapter is based on our works ([Abu Farha and Magdy, 2020](#); [Abu Farha et al., 2021](#)):

- Ibrahim Abu Farha and Walid Magdy. 2020. From Arabic Sentiment Analysis to Sarcasm Detection: The ArSarcasm Dataset. In Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, pages 32–39, Marseille, France. European Language Resource Association.
- 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, pages 296–305, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

3.1 Overview

Work on subjective language analysis has been prominent in the literature during the last two decades. A major theme that dominated the area is the work on sentiment analysis (SA). According to [Liu \(2012\)](#), SA is a process where we extract and analyse the emotional polarity in a given piece of text. A large amount of work focused on classifying the text into its sentiment class, which varies based on the granularity. SA is one of the research areas within the larger natural language processing (NLP) field. The interest in SA research was embarked by the advent of user-driven platforms such as social media websites. Research on SA started with the early work of [Pang et al. \(2002\)](#), where they analysed the sentiment in movie reviews. Since then, the work has developed and spanned different topics and fields, such as social media analysis, computational social science and others. Most of the work is focused on English, whereas Arabic did not receive much attention until after 2010. The work on Arabic SA was kicked off by [Abdul-Mageed et al. \(2011\)](#), but it still lags behind the progress in English. This can be attributed to the many challenges of Arabic language; including the large variety in dialects ([Habash, 2010](#); [Darwish and Magdy, 2014](#)) and the complex morphology of the language ([Abdul-Mageed et al., 2011](#)).

As the work on SA systems developed, researchers started analysing the intricacies of such systems in order to understand their performance and where they fail. There are many challenges when doing SA, such as negation handling, domain dependence, lack of world knowledge and sarcasm ([Hussein, 2018](#)). Sarcasm can be defined as a form of verbal irony that is intended to express contempt or ridicule ([Joshi et al., 2017](#)). Sarcasm is correlated with expressing an opinion in an indirect way, where the intended meaning is different from the literal one ([Wilson, 2006](#)). Additionally, sarcasm is highly context-dependent, as it always takes part between parties where shared knowledge exists. Usually, a speaker will not use sarcasm unless he/she thinks that it will be understood as so ([Joshi et al., 2017](#)).

Sarcasm detection is a crucial task for SA. The reason for this is that a sarcastic utterance usually carries a negative implicit sentiment, while it is expressed using positive expressions. This contradiction between the surface sentiment and the intended one creates a complex challenge for SA systems ([Bouazizi and Ohtsuki, 2016](#)).

There have been many works on English sarcasm detection, those include datasets such as (Abercrombie and Hovy, 2016; Barbieri et al., 2014a,b; Filatova, 2012; Ghosh et al., 2015; Joshi et al., 2016a) and detection systems such as (Rajadesingan et al., 2015; Joshi et al., 2015; Amir et al., 2016).

Up to the start of this work, work on Arabic sarcasm was limited to the work of Karoui et al. (2017), a shared task on irony detection (Ghanem et al., 2019) along with the participants' submissions and a dialectal sarcasm dataset by Abbes et al. (2020). Additionally, there were no publicly available datasets for Arabic sarcasm detection. The data in (Karoui et al., 2017) was not publicly available and most of the tweets provided in (Ghanem et al., 2019) were deleted.

3.2 ArSarcasm Dataset

In this work, we present ArSarcasm, a new dataset for Arabic sarcasm detection¹. The dataset consists of a combination of Arabic SA datasets, where we re-annotated them for sarcasm. In addition to that, we also provide labelling for the dialect and sentiment.

3.2.1 Resources

In this work, we relied on a set of well-known Arabic SA datasets. The reason for this choice is that sarcasm is highly subjective and always mentioned as one of the main reasons that degrade sentiment analysers' performance. The datasets we are using are SemEval's 2017 (Rosenthal et al., 2017) and ASTD (Nabil et al., 2015) datasets. ASTD dataset consists of 10,006 tweets labelled as shown in Table 3.1. The dataset contains tweets that date back to the period between 2013 and 2015. The tweets are mostly in Egyptian dialect and they were annotated using Amazon's Mechanical Turk. In our work, since we are aiming to annotate for sarcasm, we decided to eliminate the objective class and we took our sample from the other subjective classes.

Class	Positive	Negative	Neutral	Objective	Total
Count	799	1,684	832	6,691	10,006

Table 3.1: ASTD statistics.

¹ArSarcasm is available at: <https://github.com/iabufarha/ArSarcasm>

The other dataset we are using is the one provided in SemEval’s 2017 task for Arabic SA (Rosenthal et al., 2017). This dataset consists of 10,126 tweets distributed over different sets, as shown in Table 3.2. The data was annotated using CrowdFlower² crowd-sourcing platform. The new dataset contains 10,543 tweets, most of which were taken from SemEval’s dataset.

Set	Positive	Negative	Neutral	Total
Training	743	1,142	1,470	3,355
Validation	222	128	321	671
Testing	1,514	2,222	2,364	6,100
Total	2,479	3,492	4,155	10,126

Table 3.2: SemEval 2017 Task 4-A dataset statistics.

3.2.2 Annotation

For the annotation process, we used Appen crowd-sourcing platform. Our main objective was to annotate the data for **sarcasm** detection, but due to the challenges imposed by dialectal variations, we decided to add the annotation for **dialects**. We also include a new annotation for **sentiment** labels in order to have a glimpse of the variability and subjectivity between different annotators. Thus, the annotators were asked to provide three labels for each tweet as the following:

- **Sarcasm:** sarcastic or non-sarcastic.
- **Sentiment:** positive, negative or neutral.
- **Dialect:** Egyptian (Nile Basin), Gulf, Levantine, Maghrebi or Modern Standard Arabic (MSA).

To keep the sentiment annotation process consistent, we used the same guidelines that were used to annotate SemEval’s dataset. Regarding sarcasm, we define it as *an utterance that is used to express ridicule, where the intended meaning is different from the apparent one*.

Only annotators who have Arabic language in their profiles and come from an Arab country were allowed to participate. Each tweet was annotated by at least *three*

²Currently Appen, Figure-Eight before that.

different annotators. The quality of annotation was monitored using a set of 100 hidden test questions that appear randomly during the task; each of those questions has the correct label for sentiment, sarcasm and dialect. If the performance of an annotator in these test questions drops below 80%, this annotator is eliminated and all the labels he/she provided are also ignored. In such a case, new annotators will be asked to provide labels to replace the ignored ones. The final label is based on the majority vote. Since the sarcasm label is binary and we are collecting an odd number of annotations, three, we will always have a majority. Agreement³ among annotators was 80.7% for sentiment, 89.3% for sarcasm and 86.7% for dialects. A sample of the guidelines provided to the annotators is shown in Appendix A.

3.3 Statistics and Analysis

3.3.1 Dataset Statistics

The new dataset contains 10,547 tweets, 8,075 of them were taken from SemEval's dataset, while the rest (2,472 tweets) were taken from ASTD. Each tweet has three labels for sarcasm, sentiment, and dialect. Table 3.3 shows the statistics of the new dataset, where we can see that 16% of the data is sarcastic (1,682 tweets). The new annotation shows that most of the data is either in MSA or the Egyptian dialect, while there are few examples in the Maghrebi dialect. Figure 3.1 shows the ratio of sarcasm in the tweets belonging to each dialect. Maghrebi dialect has the largest percentage, but this is an outlier due to the small number of Maghrebi tweets (only 32 tweets). Thus, sarcasm is more prominent in the Egyptian dialect with 34% of the Egyptian tweets being sarcastic. Also, from the table, it is noticeable that the Egyptian dialect comprises most of the sarcastic tweets (799 tweets, 47.5% of the sarcastic tweets). When looking at Figure 3.1 and Table 3.3, it is noticeable that most of the sarcastic sentences are in dialectal Arabic, comprising 62.5% compared to 37.5% in MSA. Also, it is noticeable that MSA has the lowest sarcasm ratio, which indicates that people tend to use MSA to convey their ideas directly. The reason for having a few examples in the Maghrebi dialect is due to the original data collection approach. The original authors used keywords of popular topics at that time (2016), and most of them are related to the Eastern part of the Arabic world. Table 3.4 provides examples of sarcastic tweets from different dialects.

³In this case, the agreement represents the average percentage of votes the majority label received.

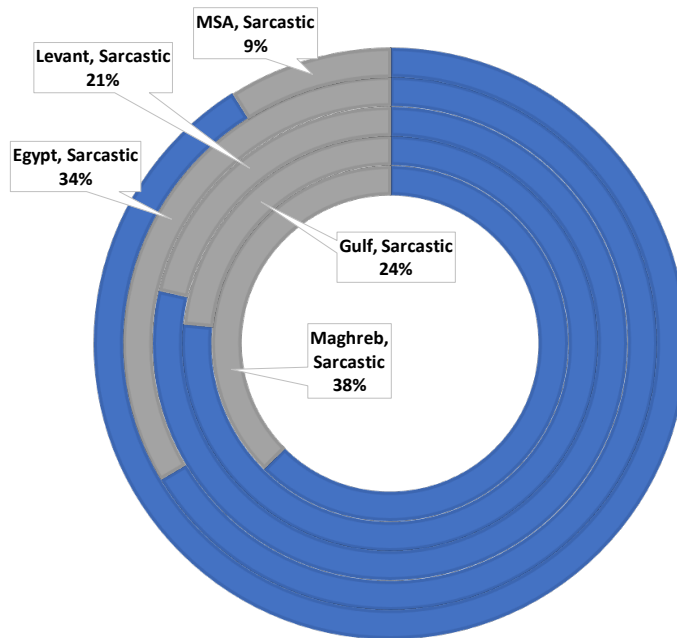


Figure 3.1: Ratio of sarcasm over the dialects.

Dialect	Non-Sarcastic	Sarcastic	Negative	Neutral	Positive	Total
Egyptian/Nile	1,584	799	1,179	733	471	2,383
Gulf	397	122	200	218	101	519
Levantine	433	118	239	178	134	551
Maghrebi	20	12	18	10	4	32
MSA	6,431	631	1,893	4,201	968	7,062
Total	8,865	1,682	3,529	5,340	1,678	10,547

Table 3.3: Dataset statistics for sarcasm and sentiment over the dialects.

3.3.2 Sentiment in Sarcasm

In this section, we analyse the relationship between sarcasm and sentiment. Figure 3.2 shows the sentiment distribution over the sarcastic tweets. It is clear that most of the sarcastic tweets have negative sentiment, and this agrees with the definition we adopted, which implies that sarcasm includes making ridicule of someone or something. However, there are some neutral and positive sarcastic tweets, which could be due to the highly subjective nature of sarcasm. In addition, this could be attributed to the fact that some other metaphoric or figurative expressions might fall under the sarcasm definition. An example of that is understatement, where a person describes a good thing using negative terms such as “This was an extremely hard exam”. This

phenomenon is demonstrated in example 2 in Table 3.4, where the speaker is bragging about his success in being a presenter, and he mentions that this had happened because his mother wished him to be embarrassed and looked at as a weird person.

Table 3.4 provides examples of sarcastic tweets from different dialects along with their sentiment. Those examples show some aspects of the sarcasm nature, such as referencing real world items or figures. The examples show how challenging sarcasm can be, as some of them are expressed using positive expressions, yet having negative sentiment and vice versa. This, in turn, makes it extremely challenging for an SA system to analyse such examples, which urges the need for sarcasm detection systems. They also show that sarcasm relies heavily on world knowledge and context; thus incorporating such information is necessary to correctly identify sarcasm.

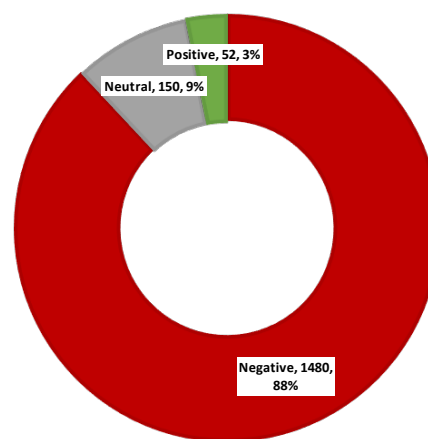


Figure 3.2: Sentiment distribution over the sarcastic tweets.

3.3.3 Annotation Subjectivity

We also studied the difference between the original and new sentiment labels. Figure 3.3 shows how the new labels are different from the original ones, labels above the charts are the original ones. It is clear that there is an extreme change in the labels. This is an empirical proof of the highly subjective nature of sentiment analysis annotation. We can see that in the case of the positive class, more the 50% of the labels have been changed, Table 3.5 provides examples of these cases. From the table, it is noticeable that these cases can be attributed to different reasons. For example, in the second tweet, the original annotator failed to perceive the sarcasm intended by the author. This can be due to either a misunderstanding of the intentions or a mismatch between the author's

ID	Tweet	Sentiment	Dialect
1	<p>كنت أعتقد أن خدمة غوغل ترجمة سيئة جدا إلى أن جربت بينغ. ملك جمال الترجمة غوغل</p> <p>LT: I was thinking that Google Translate is bad, till I tried Bing. Google is Mr. Translation. IM: Bing Translate is bad, Google's is better.</p>	Negative	MSA
2	<p>واضح إن أمي دعت عليا وأنا صغير وقالتلي روح ربنا يفرج عليك خلقوا، قام ربنا طلعتني مذيع</p> <p>LT: it is clear that my mother was mad at me and wished that I get embarrassed and looked at by people, Now I am a TV presenter. IM: I am a famous TV presenter.</p>	Positive	Egyptian
3	<p>بالصيفيات الحلوه محد يقرر ينزلني على لبنان لما وصلت درجه الحراره تحت الصفر امي تقول ن فكر نروح لا شكرا</p> <p>LT: when it is summer, no one suggests going to Lebanon. Now, when it is below zero, my mother considers going there. No, thanks. IM: I don't want to go to Lebanon in the cold winter.</p>	Negative	Levantine
4	<p>الناس المؤمنين بالسحر كان لازم نوضحلهم ان هاري بوتر مو فلم وثائقي</p> <p>LT: we should have explained for those who believe in magic that Harry Potter is not a documentary. IM: those who believe in magic, truly think that Harry Potter is real.</p>	Negative	Gulf

Table 3.4: Examples of some sarcastic tweets from different dialects. LT stands for literal translation. IM stands for intended meaning.

intention and the annotator's preference. The other reason that might have caused the labels to change is the different perspectives that a text can be looked at from. For example, some annotators might annotate news as neutral, considering the view of the news agency, while others might reflect their own preferences. The same thing occurs if the text is about two conflicting parties, where the annotators are likely to take one side. In addition to that, the available Arabic SA datasets are highly political and they contain different dividing topics. Having all of these factors together would result in the high presence of the annotator's biases and personal views.

Moreover, in the case of most sentiment and sarcasm datasets, they were annotated using crowd-sourcing platforms. These platforms provide multiple annotations for each data point, but they do not ensure having the same annotators to annotate all the data. This would provide inconsistent labels for the subjective text, where different conflicting biases are reflected on the assigned label. Thus, having multiple people annotating a dataset would probably give conflicting labels for different related instances

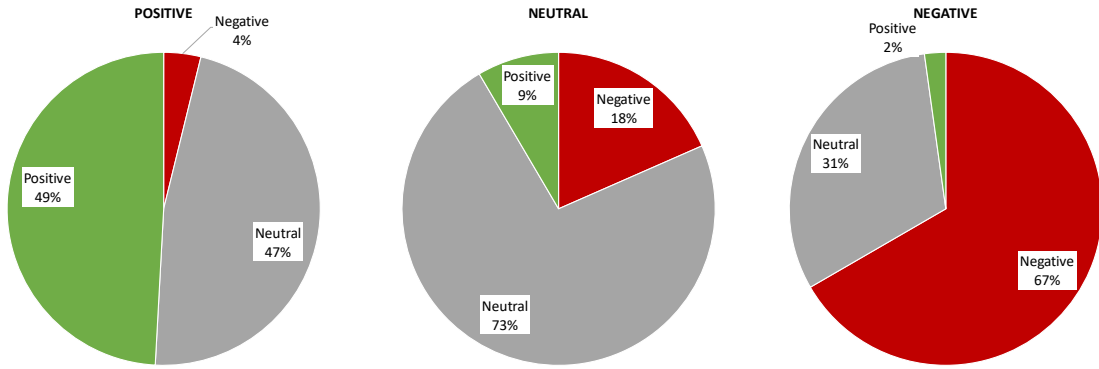


Figure 3.3: The change in sentiment labels between the original and new annotation. The labels above the charts are the original labels.

within the data. These phenomena impose challenges for sentiment analysis systems, since the boundaries between the labels are not clear.

Based on the previous statistics and examples, we can see that the current annotation schemes and procedures are not robust enough against bias, and they do not ensure consistency among different annotators. In addition, the current approach of considering sarcasm as a binary text classification problem is not precise. Sarcasm is highly related to the context, cultural background, world knowledge and personal traits of its author. We believe that more sophisticated data collection and annotation approaches should be used to have a proper computational representation of sarcasm.

ID	Tweet	Original label	New label
1	جوجل تنافس أبل وسامسونج بهاتف جديد Google is competing Apple and Samsung with a new phone	Positive	Neutral
2	مبروك عليك ويندوز ١٠ .. ربنا يخلهولللك Congratulations on Windows 10, God keekeep it for you	Positive	Negative
3	اخش مشغلين اغنيه جستن بيبير Shame, they are playing a Justin Bieber song	Neutral	Negative
4	سيتم الرد علي حضرتك في اقرب وقت يا فندم. Sir, we will respond to you soon	Neutral	Positive
5	سمعة أبل على المحك.. مشكلة حقيقية في آيفون ٧ Apple's reputation is on the line ... A real problem in iPhone 7	Negative	Neutral
6	طقس كاذب يقولو ثلوج ويطلع حر Deceitful weather, they say it will snow and it is warm	Negative	Positive

Table 3.5: Examples of some tweets that have their labels changed.

3.4 Effect of Sarcasm on Sentiment Analysis

To better understand how sarcasm can be disruptive for SA systems, we conducted an experiment on the newly annotated data. This was done through comparing the performance of an available SA system on both sarcastic and non-sarcastic tweets. In this experiment, we used Mazajak (Abu Farha and Magdy, 2019), an online Arabic sentiment analyser. In order to have an informative comparison, we separated the dataset into two sets, sarcastic (1,682) and non-sarcastic (8,865). The performance was compared using the original and new sentiment labels. Table 3.6 shows the achieved macro F1-score. It is clear that there is a gap between the performance on sarcastic and non-sarcastic tweets. Mazajak achieved F1-scores of 0.43 (new labels) and 0.44 (original labels) on sarcastic tweets, and F1-scores of 0.64 (new labels) and 0.61 (original labels) on the non-sarcastic ones.

Although Mazajak was trained on samples from the same dataset, the results on the sarcastic tweets are much lower than those on the non-sarcastic ones. The low performance on the sarcastic tweets indicates that SA systems rely mostly on the surface sentiment expressed by the words. This, in turn, means that sarcasm, which is an indirect implicit expression tool, is a major challenge for SA systems.

Set	F-score (new)	F-score (original)
Sarcastic	0.43	0.44
Non-Sarcastic	0.64	0.61

Table 3.6: Mazajak’s performance on sarcastic and non-sarcastic tweets. The references are the original and the new sentiment labels.

3.5 Arabic Sarcasm Shared Task

In this section, we provide the details of our sarcasm and sentiment detection in Arabic shared task, which is co-organised with the WANLP 2021 workshop on Arabic NLP. The goal of the shared task is to provide resources and encourage researchers to work on Arabic sarcasm detection. The shared task has two subtasks, sarcasm detection (subtask 1) and sentiment analysis (subtask 2). We provided the participant with a new dataset (ArSarcasm-v2), which is publicly available⁴. The dataset is annotated

⁴ArSarcasm-v2 is available at: <http://github.com/iabufarha/ArSarcasm-v2>

for sarcasm, sentiment and dialect. We received 27 submissions for subtask 1 and 22 submissions for subtask 2. This section provides an overview of the shared task and the achieved results by the participants along with their approaches.

3.5.1 ArSarcasm-v2 Dataset

The shared task utilises the ArSarcasm-v2 dataset, which is an extension of the original ArSarcasm dataset proposed in Section 3.2. The main reason to create this dataset and extend ArSarcasm is that ArSarcasm was publicly available and we needed to prepare a new test set. Hence, we utilised ArSarcasm to create ArSarcasm-v2.

Resources

ArSarcasm-v2 uses the whole original ArSarcasm dataset as part of its training data. The original ArSarcasm consists of 10,547 tweets, 1,682 of which are sarcastic. Additional sarcastic tweets are added to the ArSarcasm-v2 dataset from the DAICT dataset (Abbes et al., 2020), which represents a corpus of ironic/sarcastic tweets. DAICT contains 5,358 tweets, 4,809 of which are ironic/sarcastic.

Since the goal is to extend the larger ArSarcasm, and because DAICT is mostly sarcastic, a new set of random tweets were collected over the period November-December 2020. The tweets were collected using the Twitter streaming API with the language filter set to Arabic (“lang:ar”). Since sarcasm is usually present in a small percentage, the new tweets were used to balance out DAICT.

Annotation

For the annotation process, we used Appen⁵ crowd-sourcing platform. ArSarcasm represents the majority portion of ArSarcasm-v2. Thus, the goal was to annotate the new portions to have similar labels to ArSarcasm. Additionally, DAICT was only annotated for sarcasm/irony; thus a new annotation was needed. To ensure consistency with ArSarcasm, we followed the same procedure and used the same guidelines to annotate the new portions.

Since DAICT is only annotated for sarcasm/irony, it was used as a pool of sarcastic examples which were balanced with the set of random Arabic tweets. A new set of 5,000 tweets, 2,500 of which are from DAICT, were annotated. The annotators were asked to provide three labels for each tweet as the following:

⁵<https://www.appen.com/>

Dialect	Sarcastic	Non-Sarcastic	Negative	Positive	Neutral	Total
MSA	1,523	9,362	3,986	1,890	5,009	10,885
Egyptian/Nile	1,085	1,896	1,564	524	893	2,981
Gulf	214	752	411	192	363	966
Levantine	152	519	312	143	216	671
Maghrebi	15	30	25	6	14	45
Total	2,989	12,559	6,298	2,755	6,495	15,548

Table 3.7: Statistics of ArSarcasm-v2 dataset showing the distribution of sarcasm and sentiment over the dialects.

- **Sarcasm:** sarcastic or non-sarcastic.
- **Sentiment:** positive, negative or neutral.
- **Dialect:** Egyptian (Nile Basin), Gulf, Levantine, Maghrebi or Modern Standard Arabic (MSA).

Only annotators of Arab origin were allowed to participate. This was verified through their profile (usage of the Arabic language). Each tweet was annotated by at least three different annotators. The quality of annotation was monitored using a set of 100 hidden test questions that appear randomly during the task, each of those questions has the correct label for sentiment, sarcasm and dialect. If the performance of an annotator in these test questions drops below 80%, this annotator is eliminated, and all the labels he/she provided are also ignored. Agreement among annotators was 78.9% for sentiment, 87.3% for sarcasm and 77.0% for dialects.

Dataset Statistics

The new ArSarcasm-v2 dataset consists of 15,548 tweets, 10,547 of them were taken from the original ArSarcasm dataset while the rest (5,001 tweets) from DAICT and the new collection of tweets. These additional 5,001 tweets were split into two parts: 2,001 tweets were added to the original ArSarcasm to form the set of training data of 12,548 tweets, and the remaining 3000 were used as the test set, as shown in Table 3.8. Each tweet has three labels for sarcasm, sentiment, and dialect. Tables 3.8 and 3.7 show the statistics of the new dataset, where we can see that 19.2% of the data is sarcastic (2,989 tweets). Also, the annotation shows that most of the data is either in MSA or Egyptian dialect, while the Maghrebi dialect is under-represented with only 45 tweets.

Set	Sarcasm		Sentiment			Total
	Sarcastic	Non-sarcastic	Positive	Negative	Neutral	
Training	2,168	10,380	2,180	4,621	5,747	12,548
Testing	821	2,179	575	1,677	748	3,000
Total	2,989	12,559	2,577	6,298	6,495	15,548

Table 3.8: Statistics of training and testing sets, showing the number of examples for both sarcasm detection and sentiment analysis tasks.

3.5.2 Shared Task Details

This section provides an overview of the shared task, the description of the subtasks and the evaluation metrics.

3.5.3 Tasks Description

The shared task on sarcasm detection and sentiment analysis in Arabic contains two subtasks as follows:

- **Sarcasm Detection (subtask 1):** the goal is to identify whether a tweet is sarcastic or not.
- **Sentiment Analysis (subtask 2):** the goal is to classify the tweet to one of the sentiment classes: positive, negative or neutral.

The data for both subtasks was provided as train/test split without a specific development set. Table 3.8 shows the statistics of the two sets. The training set consists of 12,548 tweets, while the testing set consists of 3,000 tweets. The participants had access to the tweets' text and the dialect label during the testing phase of the task.

Evaluation Metrics

The main evaluation metric for subtask 1 (sarcasm detection) is the F1-score of the sarcastic class only ($F_1^{\text{sarcastic}}$), since it is the main class to be detected. Sarcasm is usually present in small percentages in the data, thus the task is an imbalanced classification task. $F_1^{\text{sarcastic}}$ is calculated using the following equation:

$$F_1^{\text{sarcastic}} = 2 \cdot \frac{P^{\text{sarcastic}} \cdot R^{\text{sarcastic}}}{P^{\text{sarcastic}} + R^{\text{sarcastic}}}, \quad (3.1)$$

Where $P^{\text{sarcastic}}$, $R^{\text{sarcastic}}$ are the precision and recall with respect to the sarcastic class.

Team	Affiliation of the first author	Subtask(s)
AIMTechnologies	A.I.M Technologies	1, 2
ALI-B2B-AI	Alibaba Group, China	1, 2
ArabicProcessors (Gaanoun and Benelallam, 2021)	INSEA, Morocco	1, 2
BhamNLP (Alharbi and Lee, 2021)	University of Birmingham, King Abdulaziz University	1, 2
CS-UM6P (El Mahdaouy et al., 2021)	Mohammed VI Polytechnic University, Morocco	1, 2
DeepBlueAI (Song et al., 2021)	DeepBlue Technology (Shanghai) Co., Ltd, China	1, 2
DM-JUST(dalya) (Faraj et al., 2021)	Jordan University of Science and Technology, Jordan	1
Fatemah (Faraj et al., 2021)	Kuwait University, Kuwait	1, 2
iCompass (Naski et al., 2021)	iCompass, Tunisia	1, 2
IDC (Israeli et al., 2021)	The Data Science Institute, Interdisciplinary Center, Israel	1, 2
ITAM	University Mohamed First, Oujda, Morocco	1, 2
Juha (Abuzayed and Al-Khalifa, 2021)	iWAN research group, Saudi Arabia	1, 2
Laila & Dalayah (Laila) (Bashmal and AlZeer, 2021)	King Saud University, Saudi Arabia	1
Naglaa Abdelhade (Naglaa)	Assiut university, Egypt	2
NAYEL (Nayel et al., 2021)	Benha University, Egypt	1, 2
Phonemer (Wadhawan, 2021)	Flipkart Private Limited	1, 2
rematchka (Abdel-Salam, 2021)	Computer Engineering, Cairo University, Egypt	1, 2
SalamBERT (Husain and Uzuner, 2021)	Kuwait University, Kuwait	1, 2
Serpente (Ghoul and Lejeune, 2021)	Sorbonne University, France	1, 2
SpeechTrans (Lichouri et al., 2021)	CRSTDLA Research Center, Algeria	1, 2
SPPU_AASM (Hengle et al., 2021)	Pune University, India	1, 2
ZTeam (Elgabry et al., 2021)	Helwan University, Egypt	1, 2

Table 3.9: The list of participating teams who provided their affiliation details.

For the sentiment analysis, the macro F1-score over the positive and negative classes was used (F_1^{PN}). It is worth noting that the neutral class is excluded from the metric calculation and not the whole task. Thus misclassified neutral tweets will lead to the increase of false positives for the positive or negative class, and thus should lead to the reduction of the F_1^{PN} value. This metric is the main adopted measure in multiple sentiment analysis shared tasks in different languages (Kiritchenko et al., 2016; Rosenthal et al., 2017). F_1^{PN} is calculated using the following equation:

$$F_1^{PN} = \frac{1}{2}(F_1^P + F_1^N), \quad (3.2)$$

Where F_1^P, F_1^N are the F_1 with respect to the positive and negative classes respectively, while the neutral class is ignored.

Participating Teams

The shared task saw the participation of 30 unique teams. The sarcasm detection task (subtask 1) received 27 submissions, while the sentiment analysis task (subtask 2) received 22 submissions. Table 3.9 shows the list of the participating teams and their affiliations

Shared Task Results

Tables 3.10 and 3.11 show the results of both subtask 1 and subtask 2 respectively. The results are sorted in descending order based on the official metric of the corresponding subtask, where $F_1^{\text{sarcastic}}$ and F_1^{PN} are the official metrics for subtask 1 and subtask 2 respectively. For each team, only the last submission was considered for the leaderboard. For subtask 1 (sarcasm detection), BhamNLP (Alharbi and Lee, 2021) achieved first place with an $F_1^{\text{sarcastic}}$ of 0.6225. For subtask 2 (sentiment analysis), CS-UM6P (El Mahdaouy et al., 2021) team achieved first place with an F_1^{PN} of 0.748.

Approaches by Top Submissions

The participating teams used a variety of approaches for both subtasks. Most of the teams used pre-trained language models such as AraBERT (Antoun et al., 2020) and MARBERT (Abdul-Mageed et al., 2021a). A few of the participants used deep learning and conventional machine learning approaches. All the teams, that are participating in the two subtasks, used the same architecture for both tasks.

For the sarcasm detection task, BhamNLP (Alharbi and Lee, 2021) team was ranked first with an $F_1^{\text{sarcastic}}$ of 0.6225. In their approach, they used a multi-task learning architecture that is trained for sarcasm and sentiment classification. The model is based on both MARBERT and a CNN-LSTM model, where the output of each of these models is concatenated and fed to the final output layer. The CNN-LSTM used both word and character embeddings. The second place (SPPU-AASM) (Hengle et al., 2021) used an ensemble of AraBERT and CNN-BiLSTM model, which achieved an $F_1^{\text{sarcastic}}$ of 0.614. The third place (DeepBlueAI) (Song et al., 2021) used an ensemble of AraBERT and XLM-R, which achieved an $F_1^{\text{sarcastic}}$ of 0.6127.

For the sentiment analysis task, CS-UM6P team (El Mahdaouy et al., 2021) was ranked first with an F_1^{PN} of 0.748. Their model is based on MARBERT-based Multi-Task Learning with task attention interaction layer for sarcasm and sentiment detection. The model used MARBERT as an encoder to produce sentence embeddings. Those embeddings are fed into separate attention and output layers specific to each task. The second place was DeepBlueAI (Song et al., 2021) with an F_1^{PN} of 0.7392. They used a similar architecture to the one used for sarcasm detection. The third place was rematchka (Abdel-Salam, 2021), who fine-tuned MARBERT for sentiment classification and achieved an F_1^{PN} of 0.7321.

Rank	Team	$F_1^{\text{sarcastic}}$	Accuracy	Macro-F1	Precision	Recall
1	BhamNLP	0.6225	0.7700	0.7286	0.7193	0.7460
2	SPPU-AASM	0.6140	0.7410	0.7096	0.7031	0.7447
3	DeepBlueAI	0.6127	0.7830	0.7310	0.7279	0.7345
4	CS-UM6P	0.6000	0.7680	0.7183	0.7122	0.7268
5	dalya	0.5989	0.7830	0.7251	0.7268	0.7235
6	Laila	0.5968	0.7063	0.6829	0.6874	0.7337
7	Phonemer	0.5872	0.7830	0.7200	0.7264	0.7147
8	AIMTechnolgies	0.5852	0.7467	0.7014	0.6934	0.7174
9	IDC	0.5677	0.7670	0.7041	0.7062	0.7022
10	rematchka	0.5662	0.7803	0.7095	0.7231	0.7004
11	UBC	0.5468	0.7723	0.6974	0.7119	0.6880
12	SalamBERT	0.5348	0.7727	0.6922	0.7128	0.6807
13	Juha	0.5191	0.6980	0.6495	0.6443	0.6661
14	ZTeam	0.5189	0.7533	0.6765	0.6858	0.6700
15	ALI-B2B-AI	0.5139	0.7617	0.6780	0.6965	0.6678
16	ArabicProcessors	0.5086	0.7797	0.6833	0.7296	0.6665
17	MMFOUAD	0.5056	0.6917	0.6408	0.6360	0.6557
18	Fatemah	0.5041	0.7607	0.6732	0.6950	0.6622
19	Kalawy	0.4870	0.7247	0.6494	0.6514	0.6476
20	rehab88	0.4870	0.7247	0.6494	0.6514	0.6476
21	iCompass	0.4860	0.7730	0.6702	0.7195	0.6543
22	Serpente	0.4109	0.7630	0.6313	0.7116	0.6194
23	SpeechTrans	0.3371	0.7287	0.5833	0.6359	0.5802
24	AhmedAbdou	0.2542	0.7340	0.5462	0.6486	0.5569
25	ITAM	0.2509	0.7253	0.5414	0.6218	0.5517
26	NAYEL	0.2440	0.7460	0.5457	0.7048	0.5602
27	rematchka	0.1657	0.7047	0.4932	0.5497	0.5185

Table 3.10: Results achieved by participants in subtask 1 (sarcasm detection). The main metric is the F1-score of the sarcastic class ($F_1^{\text{sarcastic}}$).

Rank	Team	F_1^{PN}	Accuracy	Macro-F1	Precision	Recall
1	CS-UM6P	0.7480	0.7107	0.6625	0.6660	0.6713
2	DeepBlueAI	0.7392	0.7037	0.6570	0.6591	0.6714
3	rematchka	0.7321	0.6957	0.6587	0.6498	0.6748
4	Phonemer	0.7255	0.6983	0.6531	0.6515	0.6623
5	IDC	0.7190	0.6923	0.6446	0.6429	0.6582
6	ArabicProcessors	0.7145	0.6817	0.6439	0.6362	0.6693
7	Juha	0.7139	0.6853	0.6297	0.6362	0.6513
8	iCompass	0.7085	0.6743	0.6423	0.6393	0.6488
9	UBC	0.7081	0.6760	0.6346	0.6274	0.6452
10	SPPU-AASM	0.7073	0.6840	0.6232	0.6421	0.6388
11	BhamNLP	0.7014	0.6753	0.6296	0.6287	0.6570
12	Fatemah	0.6877	0.6630	0.6210	0.6136	0.6318
13	AIMTechnologies	0.6850	0.6677	0.6236	0.6213	0.6263
14	ALI-B2B-AI	0.6556	0.6333	0.5955	0.5873	0.6159
15	Serpente	0.6506	0.6473	0.5784	0.5899	0.5710
16	SalamBERT	0.6259	0.6073	0.5635	0.5580	0.5813
17	ZTeam	0.6241	0.6053	0.5545	0.5578	0.5786
18	NAYEL	0.5936	0.5980	0.5291	0.5434	0.5207
19	SpeechTrans	0.5787	0.5923	0.5222	0.5321	0.5161
20	Naglaa	0.5638	0.5793	0.5158	0.5646	0.5068
21	GOF	0.4288	0.5147	0.4275	0.5764	0.4546
22	ITAM	0.3845	0.5293	0.3768	0.4054	0.3983

Table 3.11: Results achieved by participants in subtask 2 (sentiment analysis). The main metric is the macro average of the F1-scores of the positive and negative classes (F_1^{PN}).

Other Interesting Approaches

Husain and Uzuner (2021) built their model on the hypothesis that tweets with negative sentiment and tweets with sarcasm content are more likely to have offensive content. Thus, they pre-trained AraBERT (Antoun et al., 2020) on offensive language data then fine-tuned it for the target task. Israeli et al. (2021) filtered the data through down-sampling the non-sarcastic class. Their hypothesis is that the test set would be similar to the extra portions added to the original ArSarcasm. Thus, for both ArSarcasm and the added tweets, they built a topic model for each dialect and removed irrelevant topics from ArSarcasm. Additionally, they utilised a language model to augment the data with new sarcastic examples. The augmentation was done through replacing and adding new words. Finally, they fine-tuned MARBERT model (Abdul-Mageed et al., 2021a) for each dialect. Other participants used BERT models in different ways. The majority of the participants used ensemble methods, where they combined BERT-based models with other models. While most participants used the same architecture for both tasks, some participants relied on multi-task learning to train the model on both tasks simultaneously, such as in (Alharbi and Lee, 2021; El Mahdaouy et al., 2021).

3.5.4 Discussion

The shared task results presented in this section show that the performance on sarcasm detection is relatively low. This shows that the sarcasm detection task is challenging and that further work needs to be done. A general theme that can be observed is that transformer-based models achieved the highest results, especially Arabic-specific ones. This can be attributed to their ability to better represent and capture the meaning of words within a specific context. Conventional machine learning models with hand-engineered features achieved extremely low results. This matches the intuition that sarcasm is mostly about the meaning, which cannot be captured by surface features.

3.6 Summary

This chapter provides an overview of our effort to create an Arabic sarcasm dataset and answer the question “*RQ3: How to collect and reliably annotate a dataset for Arabic sarcasm detection?*”. We presented ArSarcasm, the first publicly available Arabic sarcasm dataset, which was created through the re-annotation of available Arabic sentiment datasets. The new dataset contains sarcasm, sentiment and dialect labels. The

analysis shows that sarcasm is highly prominent in sentiment datasets, with 16% of them being sarcastic which answers “*RQ1: Is sarcasm used across Arabic social media? If so, how prominent is it?*”. We also show the highly subjective nature of such datasets, which was demonstrated by the change in sentiment labels in the new annotation. To answer “*RQ2: How does sarcasm affect the performance of sentiment analysers?*”, we tested an off-the-shelf sentiment analyser on sarcastic and non-sarcastic tweets. The results on the sarcastic tweets were lower than those on non-sarcastic ones, which shows that sarcasm is disruptive for sentiment analysers. Additionally, we provided an overview of the shared task on sarcasm and sentiment detection in Arabic. The shared task utilised ArSarcasm-v2 dataset, which is an extension of ArSarcasm.

The next chapter discusses our effort to create another type of sarcasm dataset, intended sarcasm. It provides the details of the dataset and the shared task it accompanied.

Chapter 4

Intended Sarcasm Dataset

In Chapter 3, we provided the details of how we created an Arabic sarcasm dataset through manual labelling and we showed that sarcasm is present in sentiment datasets and that it can be disruptive for sentiment analysers. This chapter continues our effort in the preceding chapter to answer the third research question “*RQ3: How to collect and reliably annotate a dataset for Arabic sarcasm detection?*”. In this chapter, we provide an overview of approaches used to create sarcasm datasets, such as distant supervision and manual labelling. We highlight the issues with these approaches and how they only capture *perceived* sarcasm. We propose a new data collection method to create datasets for *intended* sarcasm. We also organise a shared task on intended sarcasm detection that utilises the new data. The shared task saw the participation of many teams, which enriches the literature with various intended sarcasm detection models and approaches. This chapter is based on our work (Abu Farha et al., 2022a):

- Ibrahim Abu Farha, Silviu Vlad Oprea, Steven Wilson, and Walid Magdy. 2022. SemEval-2022 Task 6: iSarcasmEval, Intended Sarcasm Detection in English and Arabic. In Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 802–814, Seattle, United States. Association for Computational Linguistics.

4.1 Overview

Most previous textual sarcasm detection datasets have been annotated using a **distant supervision** method. In this approach, texts are considered sarcastic if they meet pre-defined criteria, such as including specific tags (e.g. #sarcasm, #irony) (Ptáček et al., 2014; Khodak et al., 2018), or being generated by specific accounts (Barbieri et al.,

2014a). However, this can lead to noisy labels for several reasons. First, the tags may not mark sarcasm, but may constitute the subject or object of conversation, e.g. “*there is so much #sarcasm around!*”. Second, the assumption that certain tags always appear in conjunction with sarcasm, or that certain accounts always generate sarcasm (Barbieri et al., 2014a), could lead to further false positives. Third, considering those texts that do not meet the criteria as non-sarcastic is a strong assumption that can lead to false negatives.

Due to the issues outlined above, other work has relied on **manual labelling**, where sarcasm labels are provided by human annotators (Filatova, 2012; Riloff et al., 2013; Abercrombie and Hovy, 2016). As such, the labels represent *annotator perception*, which may actually differ from *author intention*. Annotators might lack awareness of the contextual devices that, as linguistic studies suggest (Grice, 1975; Sperber and Wilson, 1981; Utsumi, 2000), could be essential for clarifying the sarcastic intention of the authors.

Previous work on sarcasm detection (Van Hee et al., 2018; Ghanem et al., 2019; Ghosh et al., 2020) present datasets annotated via the two methods discussed above. The potential noisy labels that these methods can produce give us reason to be concerned about the effectiveness of models that were trained on such datasets. Recently, Shmueli et al. (2020) proposed a third method, **reactive supervision**, which aims to collect sarcastic examples based on the conversation dynamics, addressing some of these issues by using statements such as “I was being sarcastic” to automatically label texts. However, this method relies on specific cues of sarcasm which may lead to a sample that is biased toward more confusing examples that required clarification.

Further, the vast majority of sarcasm detection work (Campbell and Katz, 2012; Riloff et al., 2013; Wallace et al., 2015; Rajadesingan et al., 2015; Bamman and Smith, 2015; Joshi et al., 2016b; Amir et al., 2016; Hazarika et al., 2018; Oprea and Magdy, 2019) has focused exclusively on the English language and, due to the sociocultural aspects of sarcastic communication (Oprea and Magdy, 2020b), it is unclear if models trained on English could generalise to other languages. To our knowledge, the small amount of work on other languages such as Arabic (Karoui et al., 2017; Ghanem et al., 2019; Abbas et al., 2020) relies on either manual labelling or distant supervision. Representative of distant supervision is the work of Karoui et al. (2017), who consider Arabic equivalents of #sarcasm, such as #سخرية, #مسخرة, and #استهزاء, to collect sarcastic tweets. Other work, (Abbas et al., 2020; Ghanem et al., 2019), used either manual labelling, or a mix between manual labelling and distant supervision. When

working with Arabic data, these two labelling methods are even more problematic considering the large number of dialects of the language that vary both across and within countries. Relying on predefined tags in modern standard Arabic (MSA), such as those specified above, can thus lead to a plethora of false negatives. Similarly, the third-party annotators might be unfamiliar with the dialect of the texts they are annotating, resulting in erroneous manual labels.

4.2 Dataset

4.2.1 Overview

In light of the issues raised in the previous section and the analysis provided in (Oprea and Magdy, 2020a), we propose iSarcasmEval shared task for sarcasm detection. We introduce a new data collection method where the sarcasm labels for texts are *provided by the authors themselves*, thus eliminating labelling proxies (in the form of predefined tags or third-party annotators). For each sarcastic text, we also ask its author to rephrase the text to convey the same intended message without using sarcasm. We also include the dialect label of the text.

4.2.2 Data Collection

To collect **sarcastic texts**, we used the Appen crowdsourcing platform¹ to recruit native Arabic speakers, and we asked the participants to write a short sarcastic text on the spot. Through a pilot study, we found this on-the-spot generation approach to result in high-quality data. Figure 4.1 shows the form we used for data collection. Appendix B includes the instructions and examples used. We asked the participants to provide the following fields:

- **Sarcastic text:** text provided by the participant using sarcastic phrasing of the sentence.
- **Non-sarcastic (direct) rephrase:** text provided by the participant using explicit phrases.
- **Sarcastic phrase:** the portion of text where sarcasm is mainly expressed.
- **Explanation:** an explanation of why the author considers the text to be sarcastic.

¹<https://appen.com>

Figure 4.1: A screenshot of the data collection form.

- **Sentiment:** sentiment expressed in the original sarcastic text (Positive, Negative, Neutral).
- **Topic:** nature of the topic discussed in the text (Politics, Sports, Celebrities, Joke).
- **Dialect:** dialect used to express the previous phrases. We used the following categories: Levant (Palestine, Syria, Lebanon, Jordan), Gulf (Saudi Arabia, Yemen, UAE, Kuwait, Iraq, Bahrain, Qatar), Nile Basin (Egypt and Sudan), Maghreb (Libya, Tunisia, Algeria, Morocco).

We also gave the option to provide age and gender, which might help in conducting more detailed studies on sarcasm and its use among different age groups and genders. In order to allow for variations, each participant was asked to provide *five* sentences along with the other details. Also, each participant was allowed twice in each data collection round. All the measurements were taken to eliminate any self-identifying information about the participants. Table 4.1 shows examples of sarcastic sentences, their non-sarcastic rephrase, and an explanation of the expressed sarcasm. The provided explanation field helped us understand the author's intentions as it explains why the author assumes the sentence to be sarcastic. This was crucial when checking the

quality of the collected data and helped filter out noisy examples.

However, this methodology only provided us with sarcastic examples. As non-sarcastic examples, we used a subset of the ArSarcasm-v2 dataset, mainly those tweets that were annotated as non-sarcastic with 100% confidence, i.e. labelled non-sarcastic by all annotators.

Sarcastic text	Non-sarcastic rephrase	Explanation
<p>نافخلي نفسه وعاملي فيها جاكي شان (He is pretending to be Jacki Chan)</p>	<p>فاكر نفسه قوي (He thinks he is strong)</p>	<p>سخيرية من الشخص الذي يدعي القوة وخفة الحركة بانه مثل جاكي شان (sarcasm of the person who pretends to be strong like Jacki Chan)</p>
<p>ويندوز احسن نظام تشغيل بالدنيا (Windows is the best OS on earth)</p>	<p>ويندوز نظام تشغيل سيء (Windows is a bad OS)</p>	<p>نظام وندوز سيء و مدحه هو المسخرة (Windows is a bad OS and praising it is sarcasm)</p>

Table 4.1: Examples of sarcastic sentences from our dataset along with the rephrase that authors gave that convey the same meaning non-sarcastically and their explanation of why their sentence is sarcastic.

4.2.3 Quality Control

The data collection was run multiple times during a period of 8 months. In this stage, we managed to collect around 2,000 sarcastic sentences. During the data collection, to minimise the probability of receiving random data, we conducted the following:

- Checking that the provided text in all the fields is in Arabic.
- Checking for redundancy and overlap with already-available entries.
- Checking that none of the fields is repeated.
- Checking if the entered text contains correct Arabic words, this is done through matching with a dictionary of the most common 1M words extracted from 250M tweets.
- Checking if the sarcastic phrase is a part of the original text.
- Checking for a minimum similarity between the original sarcastic text and its non-sarcastic (direct) counterpart.
- Checking for a minimum length for the text fields.
- Each participant is allowed to provide a maximum of 2 short sentences (less than 6 words).

After manual inspection of the collected texts, we noticed that a large portion of them was not truly sarcastic, or that the non-sarcastic phrasing was not informative. In order to filter out such examples and guarantee the data quality, we hired a native speaker for each of the available dialects. Each annotator was tasked to utilise the available explanation to conduct the following :

- Filtering out sentences that are clearly non-sarcastic or uninformative.
- Identifying nonidentical rephrases and duplicates.
- Minimally correcting misspellings. This applies to similarly shaped letters that are adjacent in keyboard layouts, such as (ف, ق) and (بب, يي).
- Improving or providing the non-sarcastic rephrase. This is because some participants provided single-word rephrases for their sentences.

After this filtration process, we ended up with around 1,300 sarcastic sentences. Since the data was used for a shared task and to maintain consistency with the size of the English dataset, we did not publish the complete dataset. The statistics in the following sections represent the portion of the data used for iSarcasmEval shared task.

4.2.4 Limitations

Since sarcasm is used in a conversational setting, asking participants to provide text on-the-spot eliminates the conversational aspect. Thus, the collected data are sentences that are sarcastic in a scenario assumed/imagined by the participant. Hence, the definition of sarcasm is fluid and depends on the participant's point of view and assumed scenario. Due to the lack of a conversational aspect and since the author is the only person participating in the assumed scenario, this would make the provided sentences more personal and require more context to understand them. While sarcasm used on social media or in a conversation is meant to be understood by others, which would make its nature different from the on-the-spot (first-party) one. Additionally, in the case of third-party annotated data, the annotation process, in a way, asks the annotator whether they can imagine a scenario where the sentence is sarcastic. In the case of first-party annotation, the annotator would probably imagine the scenario and then provide the sentence. This distinction leads to slight differences in what the sarcasm label means, which should be investigated in future work.

4.3 **iSarcasmEval** Details

The new dataset was used to organise a SemEval shared task on intended sarcasm detection, namely *iSarcasmEval* (Abu Farha et al., 2022a). The shared task targeted both Arabic and English. In the following sections, we only focus on the Arabic part of the task.

4.3.1 Task Description

The shared task contains the following subtasks:

- **Subtask A - Sarcasm Detection:** Given a text, determine whether it is sarcastic or non-sarcastic;
- **Subtask B (English only) - Sarcasm Category Classification:** Given a text, determine which ironic speech categories it belongs to, if any;
- **Subtask C - Pairwise Sarcasm Identification:** Given a sarcastic text and its non-sarcastic rephrase, i.e. two texts that convey the same meaning, determine which is the sarcastic one.

Subtasks A and C are suggested for both languages. Subtask B is only suggested for English, as we only have ironic speech category labels for English texts.

4.3.2 *iSarcasmEval* Arabic Dataset

For the shared task, we only use sarcastic and non-sarcastic rephrases. The dataset is provided as a list of texts. Each text is accompanied by a sarcasm label, indicating whether or not it is sarcastic. For sarcastic texts, there is a rephrase that conveys the same message non-sarcastically. We also include a label specifying the dialect. As mentioned previously, our data collection method only provided us with sarcastic examples. As non-sarcastic examples, we used a subset of the *ArSarcasm-v2* dataset, mainly those tweets that were annotated as non-sarcastic with 100% confidence, i.e. labelled non-sarcastic by all annotators.

The training set contains 3,103 texts, 745 of which are sarcastic. The sarcastic text have their non-sarcastic phrasing too. The test sets are the same size as the English test sets for both subtasks A and C. Table 4.2 provides a summary of the Arabic dataset splits. Table 6.7 provides the distribution of the whole dataset over the available dialects. It is noticeable that the majority of the sarcastic examples are in the Egyptian dialect (Nile Basin).

split	total	sarcastic	non-sarcastic
train	3,103	745	2,358
test (subtask A)	1,400	200	1,200
test (subtask C)	400	200	200

Table 4.2: Statistics for the Arabic training set, and test sets for subtasks A and C, as discussed in Section 4.3.2.

dialect	total	sarcastic	non-sarcastic
MSA	2,035	82	1,953
Egyptian/Nile	2,072	827	1,245
Levantine	322	76	246
Gulf	278	36	242
Maghrebi (North Africa)	195	124	71

Table 4.3: Distribution of the Arabic dataset over the dialects.

4.3.3 Evaluation Metrics

The main evaluation metric for subtask A is the F1-score of the sarcastic class, referred to as $F_1^{\text{sarcastic}}$. It is computed as follows:

$$F_1^{\text{sarcastic}} = 2 \cdot \frac{P^{\text{sarcastic}} \cdot R^{\text{sarcastic}}}{P^{\text{sarcastic}} + R^{\text{sarcastic}}}, \quad (4.1)$$

Where $P^{\text{sarcastic}}$, $R^{\text{sarcastic}}$ are the precision and recall with respect to the sarcastic class, respectively.

For subtask C, the main evaluation metric is accuracy. This is appropriate since we have an equal number of sarcastic and non-sarcastic examples.

$$\text{Accuracy} = \frac{C}{N} \quad (4.2)$$

Where C is the total number of correct predictions and N is the total number of pairs of text.

4.4 Participating Teams

4.4.1 Overview

The shared task saw the participation of 60 unique teams. The most popular task was subtask A (sarcasm detection) with 43 participants for English and 32 for Arabic. Subtask C received 16 submissions for English and 13 for Arabic. The following sections provide an overview of the top teams' approaches for the Arabic subtasks.

4.4.2 Subtask A (Sarcasm Detection) - Arabic

As mentioned previously, the main metric for subtask A is the F-score of the sarcastic class. Table 4.4 shows the results for Arabic. The participating teams made extensive use of Arabic pre-trained language models such as MARBERT (Abdul-Mageed et al., 2021a).

As shown in the Table 4.4, the top team CS-UM6P (El Mahdaouy et al., 2022) achieved an $F_1^{\text{sarcastic}}$ of 0.563. This team utilised a transformer encoder (MARBERT), attention layer, and a classifier. They applied the attention to the contextualised embeddings. The classifier, which is composed of one hidden layer, is fed the concatenation of the pooled output of the encoder and the attention's output. The official submission was an ensemble of two variants of this model that are trained with and without the non-sarcastic rephrasing. AlexU-AL (Lotfy et al., 2022) achieved the second place with an $F_1^{\text{sarcastic}}$ of 0.508. Their model is based on fine-tuning MARBERT for text classification.

The third team, remarkhka (Abdel-Salam, 2022), also used MARBERT in a similar way to AlexU-AL team. Their results are quite close to the other two models with $F_1^{\text{sarcastic}}$ of 0.477. The other teams followed a similar approach where they utilise one of the many flavours of Arabic-specific models or the multilingual ones. A few of the participants relied on hand-engineered features along with conventional classifiers such as SVM and Decision Trees.

4.4.3 Subtask C (Pairwise Sarcasm Identification) - Arabic

Table 4.5 shows the results of this task. The top team, Naive, achieved an accuracy of 0.930. They utilised the model created for subtask A, where they would compare the probabilities for each sentence and choose the one with a higher probability. Their

r	Team Name	Affiliation	$F_1^{\text{sarcastic}}$
1	CS-UM6P	Mohammed VI Polytechnic University, Morocco	0.563
2	AlexU-AL	Alexandria University, Alexandria, Egypt	0.508
3	rematchka	Cairo University, Egypt	0.477
4	HIGH-TECH Team	High Technology School, Morocco	0.468
5	Naive	Dalian University of Technology, China	0.461
6	akaBERT	Helwan University, Egypt	0.444
7	SarcasmDet	Jordan University of Science and Technology	0.431
8	Alexa	Open-Insights, Tarjamah	0.420
9	X-PuD	Baidu & Shanghai Pudong Development Bank, China	0.419
10	Plumeria	Indian Institute of Technology Kanpur, India	0.407
11	niksss	-	0.400
12	MaChAmp	IT University of Copenhagen, Denmark	0.396
13	underfined	Ping An Life Insurance Company of China, China	0.378
14	BFCAl	Benha University	0.375
15	AM	Alexandria University, Egypt	0.369
16	cnxup	University of Chinese Academy of Sciences, China	0.367
17	stce	PALI Inc., China	0.367
18	NULL	Auburn University, USA	0.358
19	Dartmouth	Dartmouth College, USA	0.350
20	Amrita-CEN	Amrita Vishwa Vidyapeetham, India	0.349
21	YNU-HPCC	Yunnan University, China	0.323
22	UMUTeam	Universidad de Murcia, Spain	0.318
23	connotation_clashers	University of Tübingen, Germany	0.301
24	LEV	Jerusalem College of Technology, Israel	0.295
25	NARD@KGP	IIT Kharagpur, India	0.281
26	JCT	Jerusalem College of Technology, Israel	0.257
27	MACHON	Jerusalem College of Technology, Israel	0.256
28	iaf7	-	0.229
29	TechSSN	Sri Sivasubramaniya Nadar College of Engineering, India	0.229
30	Sarcastic weeps	FAST NUCES LHR, Pakistan	0.192
31	MarSan_AI	Part AI Research Center, Iran	0.188
32	UoR-NCL	University of Reading, UK	0.115

Table 4.4: Subtask A (Arabic) results in descending order according to the main metric ($F_1^{\text{sarcastic}}$). The table shows the teams' names, rank, affiliation, and score.

r	Team Name	Affiliation	Accuracy
1	Naive	Dalian University of Technology, China	0.930
2	HIGH-TECH Team	High Technology School, Morocco	0.885
3	MarSan_AI	Part AI Research Center, Iran	0.875
4	Plumeria	Indian Institute of Technology Kanpur, India	0.870
5	X-PuDu	Baidu & Shanghai Pudong Development Bank, China	0.840
6	rematchka	Cairo University, Egypt	0.800
7	CS-UM6P	Mohammed VI Polytechnic University, Morocco	0.780
8	YNU-HPCC	Yunnan University, China	0.755
9	AlexU-AL	Alexandria University, Alexandria, Egypt	0.745
10	Dartmouth	Dartmouth College, USA	0.680
11	NARD@KGP	IIT Kharagpur, India	0.665
12	Sarcastic weeps	FAST NUCES LHR, Pakistan	0.465
13	MaChAmp	IT University of Copenhagen, Denmark	0.200

Table 4.5: Subtask C (Arabic) results in descending order according to the main metric (accuracy). The table shows the teams’ names, ranks, affiliations, and scores.

model in subtask A relied on the voting of a 5 folds cross-validation of a BERT-based model. High-Tech team (Alami et al., 2022) achieved the second place with an accuracy of 0.885. They fine-tuned AraBERT (Antoun et al., 2020) on the concatenation of the sarcastic sentence and its non-sarcastic phrasing. The third team, MarSan_AI (Najafi and Tavan, 2022), achieved an accuracy of 0.875. Their model consisted of a T5 encoder (Raffel et al., 2020a) followed by a transformer and Bi-LSTM, the output of the Bi-LSTM is fed to an attention layer followed by a fully connected layer. The final prediction is the softmax of the output from the fully connected layer. The other teams followed the same trend where they utilised the models from subtask A for this task. Most of these models are transformer-based models such as MARBERT and AraBERT.

4.4.4 Discussion

The sarcasm detection results (subtasks A) match those in Section 3.5, where the top models were transformer-based ones. However, the performance is still relatively low, which confirms the challenging nature of the sarcasm detection task. The fact the Arabic-specific models were better than multilingual ones indicates their better ability to represent the meaning of sentences. Models that incorporated dialectal data, such as AraBERT and MARBERT, achieved higher performance which can be due to the fact

that most of the dataset is dialectal Arabic, and seeing such data during pre-training would provide better meaning representation.

For the sarcasm identification task (subtask C), the performance is relatively higher than the performance on sarcasm detection. The gap in performance between the two tasks suggests that sarcasm identification (subtask C) is easier than sarcasm detection (subtask A). This could be due to the availability of the sarcastic and non-sarcastic pairs in subtask C, which could have provided additional signals that the models utilised to achieve better performance.

4.5 Summary

This chapter provides an overview of our effort to create a dataset for intended sarcasm. It adds to our effort to answer “*RQ3: How to collect and reliably annotate a dataset for Arabic sarcasm detection?*”. The data was collected by asking people to provide and label their own words as sarcastic or not, hence intended sarcasm. This approach mitigates the issues when using other approaches. This data was used to organise SemEval-2022 task 6, iSarcasmEval. In the shared task, we introduce two new datasets for sarcasm detection in English and Arabic. iSarcasmEval contains three subtasks: sarcasm detection, sarcasm category classification, and pairwise sarcasm identification. The task was quite popular with the participation of around 60 teams. Finally, we provide a high-level overview of the approaches of top teams in each of the subtasks.

In the next chapter, we explore the performance of state-of-the-art models on both intended and perceived sarcasm detection tasks, using iSarcasmEval and ArSarcasm-v2 datasets respectively.

Chapter 5

Arabic Sarcasm Detection, Modelling and Experiments

So far, in the previous chapters, we explored various approaches to create sarcasm datasets. Some of the proposed datasets were used to organise shared tasks, which helped bring attention to the Arabic sarcasm detection task and enrich the literature with various detection models and analyses. In this chapter, we extend the work discussed in previous chapters and add a large set of experiments using state-of-the-art models. We evaluate the performance of 24 models on the ArSarcasm-v2 and iSarcasmEval datasets, thus benchmarking the models for both perceived and intended sarcasm. In this chapter, we try to answer the fourth research question:

- **RQ4:** What is the performance of the existing NLP approaches on Arabic sarcasm detection and how to improve it?

Which includes the following questions:

- **RQ4.1:** How do current state-of-the-art models perform on the Arabic sarcasm detection task?
- **RQ4.2:** Is there a difference between the performance on perceived sarcasm (ArSarcasm-v2) compared to intended sarcasm (iSarcasmEval)?
- **RQ4.3:** How can we improve the performance of sarcasm detection models?

Our results show that sarcasm detection is a challenging task, which manifested in the low scores. However, intended sarcasm is yet more challenging with even lower scores. Our analysis shows that models achieving the best performance are those trained only on Arabic data that include dialectal Arabic and use a larger number of

parameters. This chapter answers RQ4.1, RQ4.2, and partially RQ4.3. Chapter 6 includes more analyses and adds to the answer to RQ4.3. This chapter is based on our work (Abu Farha and Magdy, 2021):

- Ibrahim Abu Farha and Walid Magdy. 2021. Benchmarking Transformer-based Language Models for Arabic Sentiment and Sarcasm Detection. In Proceedings of the Sixth Arabic Natural Language Processing Workshop, pages 21–31, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

5.1 Overview

In recent years, the development of contextualised language representations led to a revolution in the natural language process (NLP) field. Early work on representing language started with pre-trained word representations such as Word2Vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014) and fastText (Bojanowski et al., 2017). However, these word representations were static and did not rely on the context in which they appear. Other works tried to address this issue with contextualised word representations such as ELMo (Peters et al., 2018). Howard and Ruder (2018) proposed ULMFiT, which is a method for transfer-learning that can be applied to any task in NLP. The introduction of BERT (Devlin et al., 2019) revolutionised the research on NLP. BERT is based on complicated neural network models, namely transformers. The utilisation of BERT led to achieving state-of-the-art results on many NLP tasks such as sentence completion, question answering and sentiment analysis. The advantage of BERT and similar models is that they are pre-trained on large amounts of data, which the model utilises to learn a representation of the language. Following BERT, many other transformer-based language models were released such as ELECTRA (Clark et al., 2020), GPT-1/2/3 (Radford et al., 2019; Brown et al., 2020) and T5 (Raffel et al., 2020b). In the context of Arabic, AraBERT (Antoun et al., 2020) was the first Arabic-specific transformer-based language model. The introduction of AraBERT helped improve the performance in many Arabic NLP tasks. A large set of transformer-based Arabic language models has been released. These include BERT based models such as the new large version of AraBERT (Antoun et al., 2020), QARiB (Chowdhury et al., 2020), ARBERT/MARBERT (Abdul-Mageed et al., 2021a). Also, Arabic variants of other models were released, such as AraGPT2 (Antoun et al., 2021b), Ara-ELECTRA (Antoun et al., 2021a) and Arabic ALBERT (KUIS-AI-Lab). These models vary in their architectures, sizes and the nature of their training data. While most of

these models were trained on modern standard Arabic (MSA) data; some of them, such as MARBERT, included dialectal Arabic in their training data.

In this chapter, we thoroughly compare the effectiveness of 24 Arabic language models for the Arabic sarcasm detection tasks, perceived and intended sarcasm. We test these models on ArSarcasm-v2 dataset, which includes perceived sarcasm and sentiment labels, and iSarcasmEval, which represents the intended sarcasm.

5.2 Experimental Setup

5.2.1 Datasets

In the experiments, we use ArSarcasm-v2 dataset, which was proposed in Chapter 3. ArSarcasm-v2 contains the labels for sarcasm, sentiment, and dialect. Since the sarcasm labels were collected from third-party annotators, the labels represent perceived sarcasm. In the experiments, we benchmark the models for both sarcasm and sentiment tasks. We also use the Arabic iSarcasmEval dataset, proposed in Chapter 4. iSarcasmEval’s labels were provided by the texts’ authors, thus representing intended sarcasm.

5.2.2 Models

This section provides the details of the models used in the experiments. Each of these models has been fine-tuned for sentiment classification and sarcasm detection. Fine-tuning is done through adding a fully connected layer on top of the pre-trained model. After that, each model is fine-tuned for the specific task. In this work, we compare the following models:

- **Bi-LSTM**: a baseline model based on a BiLSTM combined with Mazajak word embeddings (Abu Farha and Magdy, 2019).
- **mBERT**: multilingual BERT provided by (Devlin et al., 2019). This model is based on BERT-base architecture and was trained on data from the Wikipedia dumps of 104 languages.
- **GigaBERT**: provided by (Lan et al., 2020a). The model was trained on a large set of Arabic news articles. The training data was augmented with English translations to improve cross lingual performance.
- **XLM-RoBERTa (XLM-R)**: multilingual extension of the original RoBERTa model (Liu et al., 2019) provided by (Conneau et al., 2020). We use two variants

of this model, XLM-R-base and XLM-R-large.

- **AraBERT**: Arabic-specific BERT provided by (Antoun et al., 2020). We use all the versions of AraBERT (v0.1/1/0.2/2). AraBERT (v0.2/2) models are trained on more data compared to AraBERT (v0.1/1). We experiment with all the variants of these models (base and large) and the models with and without Farasa (Abdelali et al., 2016) pre-segmentation. AraBERT (v0.1/1) was trained on 23GB of text while AraBERT (v0.2/2) was trained on 77GB of text.
- **AraELECTRA**: Arabic-specific ELECTRA provided by (Antoun et al., 2021a). ELECTRA contains two modules, a generator and a discriminator. Usually, the discriminator is taken and fine-tuned for downstream tasks. In this work, we experiment with both the generator and the discriminator. AraELECTRA was trained on the same 77GB of text used for AraBERT.
- **Arabic BERT**: provided by (Safaya et al., 2020). The model was trained on 95GB of text from the Arabic version of the unshuffled OSCAR corpus (Ortiz Suárez et al., 2020) and the Arabic Wikipedia. The model is available in two variants based on the number of parameters (base and large).
- **Arabic ALBERT**: provided by (KUIS-AI-Lab). An Arabic version of ALBERT (Lan et al., 2020b). This model was trained on data from the Arabic version of the unshuffled OSCAR corpus (Ortiz Suárez et al., 2020) and the Arabic Wikipedia. There are three variants of this model based on the number of parameters (base, large, xlarge).
- **ARBERT/MARBERT**: provided by (Abdul-Mageed et al., 2021a). These models are based on the BERT-base and trained on a set of books and news articles. ARBERT was trained on 66GB of text only from news articles. MARBERT was trained on a larger dataset (128GB), which consists of tweets. The variation in MARBERT's training data gives it the ability to better handle the variations in dialectal Arabic, which is very useful to the tasks in this paper.
- **QARiB**: provided by (Chowdhury et al., 2020). This model was trained on various sources of data including news articles and tweets.
- **AraGPT2**: Arabic-specific GPT2 provided by (Antoun et al., 2021b). AraGPT2 is a stacked transformer-decoder model trained using the causal language modelling objective. The model was trained on 77GB of Arabic text (same as AraELECTRA and AraBERT). AraGPT2 comes in four variants: AraGPT2-base, AraGPT2-medium, AraGPT2-large and AraGPT2-mega. We experiment with the base, medium and large variants.

A summary of details about these models and their variants is shown in Table 5.1.

Model	Text nature	Lang	Text size	Variant	Number of parameters
mBERT	W	multi	N/A	-	110M
GigaBERT	W, N, OC, CS	Ar-En	N/A	-	125M
XLM-R	CC	multi	N/A	base	270M
				large	550M
AraBERT-v1	W, N, OC	Ar	23GB	base (v01)	136M
				base (v1)*	136M
				base (v02)	136M
AraBERT-v2	W, N, OC	Ar	77GB	base (v2)*	136M
				large (v02)	371M
				large (v2)*	371M
AraELECTRA	W, N, OC	Ar	77GB	discriminator	135M
				generator	60M
Arabic BERT	W, OC	Ar	95GB	base	110M
				large	340M
Arabic ALBERT	W, OC	Ar	N/A	base	N/A
				large xlarge	N/A
MARBERT	T		128GB	-	163M
ARBERT	W, N, OC, B	Ar	61GB	-	163M
QARiB	N, T, S		N/A	-	N/A
AraGPT2	W, N, OC	Ar	77GB	base	135M
				medium large	370M 792M

Table 5.1: Summary of the details of each model. The table includes details about the model variant, architecture, training data size, training data nature, number of parameters. In text nature types are: tweets (T), Wikipedia (W), news (N), OSCAR corpus (OC), Common Crawl (CC), subtitles (S), and books (B). (CS) stands for code-switching, and (*) indicates that Farasa segmentation is applied to the text. Finally, (N/A) indicates that information is not available.

5.2.3 Hyper-parameters and evaluation

In the experiments¹, we relied on the implementation provided by HuggingFace’s Transformers library (Wolf et al., 2019). We used the provided *AutoModelForSequenceClassification* which matches each model to the proper implementation. We trained the models for 5 epochs with a learning rate of $5e-6$. The maximum sequence length was set to 128 tokens. For AraBERT experiments, we used the provided pipeline, which uses Farasa (Abdelali et al., 2016) segmentation for some models. For the BiLSTM model, we used 128 hidden units, Rectified Linear Unit (*ReLU*), and *Adam* (Kingma and Ba, 2015) optimiser with a learning rate of 0.0001.

For evaluation, we used the official metrics used in the WANLP 2021 shared task on sarcasm and sentiment detection in Arabic. For the sarcasm detection task, F1-score of the sarcastic class is the main metric. For sentiment analysis, the macro average of the F1-scores of the positive and negative classes (F_1^{PN}) is the main metric. This metric, F_1^{PN} , has been adopted as the standard metric for sentiment classification in SemEval tasks (Rosenthal et al., 2017).

5.3 Results

Table 5.2 shows the results achieved by all the models on iSarcasmEval and ArSarcasm-v2 dataset. The results on the sentiment analysis task for ArSarcasm-v2 are available in Appendix C. The first five rows include the BiLSTM baseline along with the multilingual BERT models. The rest of the table contains the results achieved by Arabic-specific language models. As can be seen in Table 5.2, the results are relatively low. The low results show that sarcasm detection is a challenging task. When considering ArSarcasm-v2 dataset, the overall best model is MARBERT, which achieved an $F_1^{\text{sarcastic}}$ of 0.584. The second best models is AraBERT-large (v02), which achieved an $F_1^{\text{sarcastic}}$ of 0.579. The results on iSarcasmEval’s dataset are lower than those on ArSarcasm-v2 dataset, which suggests that detecting intended sarcasm is more challenging. The best model was QARiB, which achieved an $F_1^{\text{sarcastic}}$ of 0.540. The second best model is MARBERT, with an $F_1^{\text{sarcastic}}$ of 0.500.

From Table 5.2, it is noticeable that the use of Arabic-specific language models provides a large boost in performance. Also, larger models tend to achieve higher results due to their larger representational power. Additionally, the nature of the training data

¹All the experiments in this work were conducted on a server machine with a 32 core CPU, 512GB RAM and a Quadro RTX 6000 GPU (24GB GPU Memory).

has a significant effect on performance. Models such as MARBERT and QARiB were trained on a mixture of MSA and dialectal Arabic. It is noticeable that these models’ performance is better than similar or even larger models such as Arabic BERT-large and Arabic ALBERT-large. Finally, it is clear that monolingual models achieve higher scores than multilingual ones such as mBERT, XLM-R, and GigaBERT.

Model	ArSarcasm-v2			iSarcasmEval		
	Precision	Recall	$F_1^{\text{sarcastic}}$	Precision	Recall	$F_1^{\text{sarcastic}}$
BiLSTM	0.728	0.653	0.483	0.615	0.732	0.407
mBERT	0.683	0.622	0.425	0.562	0.625	0.321
GigaBERT	0.717	0.676	0.527	0.594	0.687	0.375
XLM-R-base	0.700	0.670	0.518	0.562	0.625	0.319
XLM-R-large	0.709	0.691	0.551	0.591	0.679	0.370
AraBERT-base (v01)	0.723	0.699	0.565	0.596	0.695	0.376
AraBERT-base (v1)	0.723	0.679	0.532	0.584	0.671	0.355
AraBERT-base (v02)	0.723	0.694	0.556	0.606	0.710	0.396
AraBERT-base (v2)	0.732	0.676	0.525	0.615	0.733	0.410
AraBERT-large (v02)	0.728	0.709	0.579	0.599	0.695	0.385
AraBERT-large (v2)	0.713	0.707	0.575	0.623	0.744	0.427
AraELECTRA (discriminator)	0.731	0.708	0.578	0.616	0.722	0.417
AraELECTRA (generator)	0.675	0.691	0.527	0.566	0.635	0.324
Arabic BERT-base	0.724	0.670	0.516	0.587	0.676	0.360
Arabic BERT-large	0.720	0.694	0.556	0.599	0.701	0.382
Arabic ALBERT-base	0.706	0.693	0.555	0.559	0.621	0.316
Arabic ALBERT-large	0.701	0.674	0.523	0.565	0.632	0.324
Arabic ALBERT-xlarge	0.705	0.678	0.530	0.585	0.674	0.357
MARBERT	0.714	0.714	0.584	0.661	0.806	0.500
ARBERT	0.729	0.709	0.578	0.595	0.692	0.377
QARiB	0.734	0.690	0.551	0.683	0.807	0.540
AraGPT2-base	0.717	0.673	0.522	0.540	0.569	0.277
AraGPT2-medium	0.697	0.673	0.522	0.580	0.662	0.348
AraGPT2-large	0.681	0.671	0.521	0.576	0.655	0.343

Table 5.2: Results achieved by all models on ArSarcasm-v2 and iSarcasmEval datasets. The table includes precision, recall, and $F_1^{\text{sarcastic}}$ scores.

5.4 Discussion

In the previous section, we provided the results of the models on both perceived and intended sarcasm tasks. It is worth mentioning that we tested SVM, which achieved an

extremely low $F_1^{\text{sarcastic}}$ of 0.139 on iSarcasmEval dataset. This poor performance suggests that traditional classifiers are not effective for the sarcasm detection task. Hence we did not include it in the results or discussion. When comparing the results shown in Table 5.2, it is noticeable that the performance on intended sarcasm (iSarcasmEval) is lower than that on perceived sarcasm (ArSarcasm-v2). The best-performing model on iSarcasmEval is QARiB which achieved an $F_1^{\text{sarcastic}}$ of 0.54. The best model on ArSarcasm-v2 is MARBERT which achieved an $F_1^{\text{sarcastic}}$ of 0.584. The mean $F_1^{\text{sarcastic}}$ for all the models on iSarcasmEval is 0.375, while on ArSarcasm-v2 is 0.537. When using a two-tailed *t-test* with $p < 0.05$, we found that the difference in performance on the two datasets is statistically significant ($p\text{-value} = 2.090 \cdot 10^{-15}$). This gap suggests that the task of intended sarcasm detection is more challenging. This could be because, in perceived sarcasm, the labels are provided by third-party annotators who might have provided their labels based on some cues present in the text, which the model could utilise. In the case of intended sarcasm, the authors provided sarcastic sentences, which can be personal and lack contextual cues. This can also be because the author provided the texts assuming a common context without it being directed to a specific audience, unlike texts in ArSarcasm-v2, which are public tweets.

The results presented in this chapter aim to provide a benchmark on both datasets and capture insights into the differences between perceived and intended sarcasm. Hence, the results mentioned in previous chapters might be higher as the aim of the experiments in this chapter was not to achieve state-of-the-art performance.

For perceived sarcasm (ArSarcasm-v2), when considering the results from the shared task in Section 3.5.3, the top team (Alharbi and Lee, 2021) achieved an $F_1^{\text{sarcastic}}$ of 0.6225. Their model was based on MARBERT and CNN-LSTM and trained in a multitask setup for sarcasm and sentiment. The second (Hengle et al., 2021) and third (Song et al., 2021) teams also used an ensemble of more than one large language model with scores of 0.614 and 0.6127, respectively. The top model in our experiments was MARBERT, with an $F_1^{\text{sarcastic}}$ of 0.580. The utilisation of more than one model seems to have provided additional information to the models, enabling them to achieve higher results than only fine-tuning a language model.

When considering iSarcasmEval results presented in Section 4.4.2, the top team (El Mahdaouy et al., 2022) achieved an $F_1^{\text{sarcastic}}$ of 0.563. Their model was an ensemble of two MARBERT-based models. The second and third teams fine-tuned MARBERT for classification and achieved $F_1^{\text{sarcastic}}$ of 0.508 and 0.447, respectively. These results are quite close to the results MARBERT achieved in our experiment of $F_1^{\text{sarcastic}}$

0.500. This difference could be due to different initialisation.

In both tasks, monolingual Arabic models trained on corpora containing dialectal Arabic, such as MARBERT and QARiB, achieved the highest results. This shows that the proximity between the task data and the models' pre-training data can affect performance, which was also demonstrated by Inoue et al. (2021). Models trained with masked language modelling (MLM), BERT variants, or replaced token detection (RTD), ELECTRA, achieved higher scores. AraGPT2 is trained using causal language modelling objective, which is useful for sentence completion and language generation tasks, but it did not perform well on the sarcasm detection task.

5.4.1 Computational Cost

The development of transformer-based language models embarked the war to develop larger and larger models with billions of parameters. This raised the question of the computational cost, the accessibility of such models and the time needed to train and fine-tune them. The largest model we experimented with is AraGPT2-Mega, which we were not able to fine-tune on the aforementioned machine due to lack of memory. Figure 5.1 shows the time needed to fine-tune each of the models for one epoch with the same batch size. From the figure, it is noticeable that the time is monotonically increasing with the number of parameters within the model. AraGPT2-large (792M parameters) is the slowest, while the fastest is Arabic ALBERT-base.

When considering both the time and performance, AraELECTRA is one of the fastest and lightest models to fine-tune, yet it achieves results closer to other larger models. Thus, it could be the choice when dealing with limited resources. In our experiments, AraELECTRA achieved results closer to larger models (AraBERT-large), while it consumes half the space in the memory and can be fine-tuned much faster.

5.5 Summary

This chapter provides a set of benchmarking experiments for state-of-the-art models on both perceived and intended sarcasm detection tasks. To answer the fourth research question, "*RQ4: What is the performance of the existing NLP approaches on Arabic sarcasm detection and how to improve it?*", we evaluate 24 state-of-the-art models on both ArSarcasm-v2 and iSarcasmEval datasets, which represent perceived and intended sarcasm respectively. The results on both datasets are relatively low.

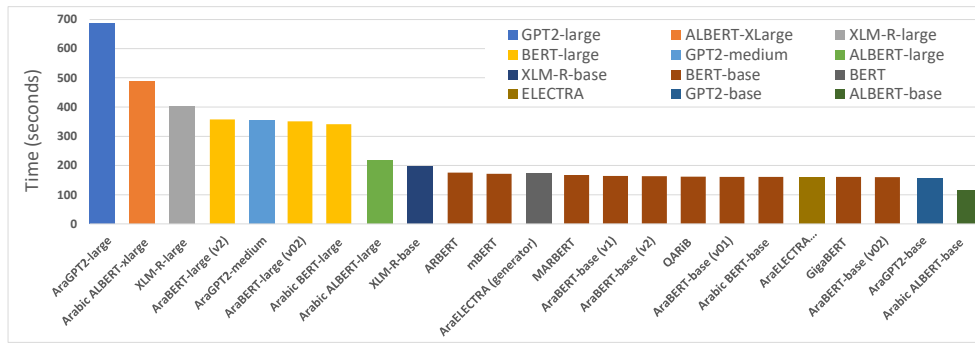


Figure 5.1: Time (in seconds) needed to fine-tune each model for one epoch for the sarcasm detection task (ArSarcasm-v2 dataset).

The best-performing model on ArSarcasm-v2 achieved an $F_1^{\text{sarcastic}}$ of 0.584, while the best model on iSarcasmEval achieved an $F_1^{\text{sarcastic}}$ of 0.540. This answers the question “*RQ4.1: How do current state-of-the-art models perform on the Arabic sarcasm detection task?*”. When comparing the results of all models on both datasets, the results on iSarcasmEval are lower than those on ArSarcasm-v2, which suggests that intended sarcasm is more challenging, which answers the question “*RQ4.2: Is there a difference between the performance on perceived sarcasm compared to intended sarcasm?*”. This chapter provides a partial answer for “*RQ4.3: How can we improve the performance of sarcasm detection models?*”. The experiments show that including social media data in pre-training would improve the performance on the tasks under study. Also, language-specific models tend to perform better than multilingual ones.

The next chapter includes an extensive error analysis of humans’ performance on sarcasm detection. We compare humans’ performance against state-of-the-art machine learning models. We provide a thorough analysis highlighting the features that make sarcasm challenging and suggesting directions for improvement, which would add to the answer of RQ4.3. We also extend the analysis to understand how Arabic dialect familiarity affects someone’s ability to understand sarcasm.

Chapter 6

Humans and Sarcasm Detection

In previous chapters, we provided a set of datasets and experiments using machine learning models. This chapter provides an in-depth analysis of the works discussed in the preceding chapters. In this chapter, we try to answer the question “*RQ5: How do Arabic speakers perform on sarcasm detection and how does dialectal variation affect that?*”, which can be divided into the following:

- **RQ5.1:** How do humans perform on sarcasm detection?
- **RQ5.2:** How does dialectal variation affect sarcasm detection?

We first start by comparing human sarcasm detection performance against machine learning models. We follow this by an in-depth error analysis highlighting the cases that were challenging for humans and/or machine. We extend this by focusing on the Arabic dialects. We analysed how dialect familiarity can affect people’s ability to understand sarcasm. Our analysis also highlights the issues that might arise when annotating dialectal Arabic data. This chapter is based on our works ([Abu Farha et al., 2022b](#); [Abu Farha and Magdy, 2022](#)):

- Ibrahim Abu Farha, Steven Wilson, Silviu Oprea, and Walid Magdy. 2022. Sarcasm Detection is Way Too Easy! An Empirical Comparison of Human and Machine Sarcasm Detection. In Findings of the Association for Computational Linguistics: EMNLP 2022, pages 5284–5295, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Ibrahim Abu Farha and Walid Magdy. 2022. The Effect of Arabic Dialect Familiarity on Data Annotation. In Proceedings of The Seventh Arabic Natural Language Processing Workshop (WANLP), pages 399–408, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

6.1 Human vs Machine

So far, we have shown that sarcasm can be disruptive for sentiment, which makes it imperative to create sarcasm detection models to mitigate for that. Doing this, however, requires reliable evaluation datasets and a further understanding of where current models (and even humans) fail on these datasets. Most of the previous sarcasm datasets have been created using either distant supervision or manual labelling. Those approaches produce unreliable labels since rule-based systems can suffer from sampling bias, and third-party annotators do not know whether the intention of the author was to truly be sarcastic. Recently, new datasets that contain first-party labels have been released. One method to collect such data is reactive supervision, where conversational cues such as “I was being sarcastic” are used as labels of previous comments (Shmueli et al., 2020). This increases reliability by identifying texts in which the authors do claim to be sarcastic, but since texts are sampled according to predefined patterns, the data may be biased toward cases that required clarification.

Yet another method for collecting first-party labels even more directly is to ask authors to provide explicit annotations of their own texts (Oprea and Magdy, 2020a). This approach eliminates annotation proxies, further reduces sampling and annotator bias, and allows for the collection of additional data about each sarcastic text, such as explanations and rephrases.

Although more difficult than earlier sarcasm detection tasks, classification models have shown promising performance on first-party annotated datasets, as evidenced in Chapter 4. However, there is currently no analysis of how *humans* would perform on these datasets. This kind of analysis can provide a range of benefits, including empirical insights into the difference in quality between third-party and first-party annotations. Additionally, most of previous works focused on building resources and detection models, without as much attention being paid to error analysis. These analyses are necessary to give insights about the limitations of the current best models, and pave the way to mitigate these limitations in the future. In this work, we aim to fill this gap and answer the following research questions:

- **RQ5.1.1:** How do humans perform on author-annotated sarcasm detection tasks?
- **RQ5.1.2:** How does human performance on these tasks compare with state-of-the-art text classifiers?
- **RQ5.1.3:** What makes sarcasm challenging for both humans and classification models?

In this work, we answer these questions by measuring both human and machine performance on iSarcasmEval’s Arabic dataset, which has first-party sarcasm labels. We make the following contributions: (1) we collect new human annotations for the iSarcasmEval dataset; (2) We analyse both humans’ and state-of-the-art (SOTA) models’ performance on Arabic sarcasm detection task, identifying cases where each succeeds and fails; (3) We analyse the error cases in order to determine the current limitations of sarcasm detection methods; and (4) we provide recommendations, based on our empirical evidence, for improving sarcasm detection models in the future.

Our analysis shows that sarcasm detection is challenging for humans, who perform nearly as well as state-of-the-art models on their own, and even better when their annotations are combined through majority voting. However, human performance using third-party labels is still imperfect and casts doubt on its utility as a source of ground truth for this task. We find that context and world knowledge are necessary to understand sarcasm in many cases. Thus, future works on sarcasm should focus on including this kind of information into datasets and leveraging it in detection models.

6.1.1 Analysis of Sarcasm Detection

It is clear that most of the work in this area has focused on how to improve data quality and how to mitigate the issues that would arise when using a specific approach. However, the literature lacks extensive analysis of which types of examples are easiest and most difficult to make accurate predictions about, yet there is some work in this direction. Some work has focused on analysing the effect of including context in sarcasm detection models (Wallace et al., 2014; Abercrombie and Hovy, 2016; Oprea and Magdy, 2019). Wallace et al. (2014) showed that annotators tend to need context to provide judgements about ironic content. They showed that there is a correlation between that and the misclassified cases. Oprea and Magdy (2019) explored the effect of contextual information to detect sarcasm, and Oprea and Magdy (2020b) analysed the effect of cultural background and age on sarcasm understanding. Their analysis indicates that age, English language nativeness, and country are significantly influential on sarcasm understanding and should be considered in the design of sarcasm detection systems. Similar results were confirmed in the case of spoken sarcasm, where Puhacheuskaya and Järvikivi (2022) found that having a foreign accent had a negative impact on irony understanding.

6.1.2 Methodology

Dataset

In this work, we use SemEval-2022 Task 6, iSarcasmEval dataset, introduced in Chapter 4. The shared task includes three subtasks: (1) sarcasm detection (subtask A): given a text, determine whether it is sarcastic or non-sarcastic; (2) sarcasm category classification (subtask B): given a piece of text, determine which ironic speech categories it belongs to; and (3) pairwise sarcasm identification (subtask C): given a sarcastic text and its non-sarcastic rephrase, determine which is the sarcastic one. Subtasks A and C cover both English and Arabic, while subtask B is English only. In this work, as we aim to analyse the performance on Arabic, we use the test sets of subtasks A and C. The test set for Task A consists of 1400 examples, while the set for task C consists of 200 pairs, each containing a sarcastic text and its non-sarcastic rephrase, written by the same author.

Human Annotation

To analyse human performance, we decided to measure how humans would perform on the test sets and compare that to the performance of computational models that participated in the shared task. To this end, we collected human annotations for the test sets using Appen¹ crowdsourcing platform, which is the same platform used to create the iSarcasmEval dataset.

We collected 5 annotations for each item². We allowed only native Arabic speakers to participate. Before starting the annotation process, each annotator is presented with test questions and only those who answer all the questions correctly would be allowed to participate in the annotation process. The test questions were sampled from a set of sentences that are clearly sarcastic/non-sarcastic. We used this approach to make sure that the annotators are not giving random answers and to avoid introducing any bias before the annotation. The average percentage of votes that the majority label received for tasks A and C are 88% and 94%, respectively.

¹<https://appen.com>

²The data is available at: <https://github.com/iabufarha/iSarcasmEval>

6.1.3 Results and Analysis

In this subsection, we compare the performance of humans against state-of-the-art models. We consider comparing the human predictions with the top-performing system for each subtask, as well as with a combination of the top five performing systems using majority voting. However, in all cases, we find that using the output from the single top team for the subtask outperformed the combination of the top five. Therefore, we only compare the human predictions with the single³ best performing model in each subtask.

For subtask A (Arabic), [El Mahdaouy et al. \(2022\)](#) were ranked first with an $F_1^{\text{sarcastic}}$ of 0.563. They used an ensemble of models based on MARBERT ([Abdul-Mageed et al., 2021a](#)). For subtask C, the top team achieved an accuracy of 0.930. Their model is based on Arabic BERT ([Safaya et al., 2020](#)).

General performance

Table 6.1 shows the general performance on both tasks. From the table, it is noticeable that when taking the majority vote from the human annotators, the performance on both tasks is better than the models submitted by the top team who participated in the respective shared task.

The scores achieved by humans when considering individual annotations, rather than majority vote, would have achieved second place in both tasks, indicating how challenging the tasks are, even for humans. When conducting McNemar’s test ([McNemar, 1947](#)), the results show that the error distributions of humans and the machine model are different except for the Arabic subtask C (pairwise sarcasm identification). A deeper look into the nature of these errors is in the following subsection.

Annotation	Task A ($F_1^{\text{sarcastic}}$)	Task C (Acc)
Human (majority vote)	0.665	0.935
Human (individual-level)	0.525	0.909
Machine (SOTA)	0.563	0.930

Table 6.1: Results for humans with majority voting, humans individually, and the top performing system for both the sarcasm detection (task A) and pairwise identification (task C). $F_1^{\text{sarcastic}}$ is the $F1$ score for the sarcastic class, the official metric used in the shared task.

³Although these are already, in some cases, ensembles of several other models.

Performance analysis

Table 6.2 shows the annotation agreement (Cohen’s Kappa) between humans’ majority label and the top team in the respective task. These results demonstrate that although both human majority voting and the state-of-the-art machine-based methods achieved similar performance, they only have moderate agreement with one another for task A. On the other hand, there was substantial agreement between the human and machine annotations for task C. Based on these preliminary results, in this subsection, we examine cases where sarcasm was detected by humans and/or machines in order to further investigate both the differences and similarities between the sets of annotations. For the analysis, as mentioned previously, we consider humans’ majority vote vs top team.

Task	Arabic
Task A	0.49
Task C	0.77

Table 6.2: Cohen’s kappa agreement between the human (majority vote) annotations and the predictions from the top performing system in the respective task.

Figure 6.1 shows the quantitative difference between human and machine generated labels for task A. We can observe that most of the generated labels are correct. The machine model was better than humans at identifying sarcastic text, while humans were better at identifying non-sarcastic text.

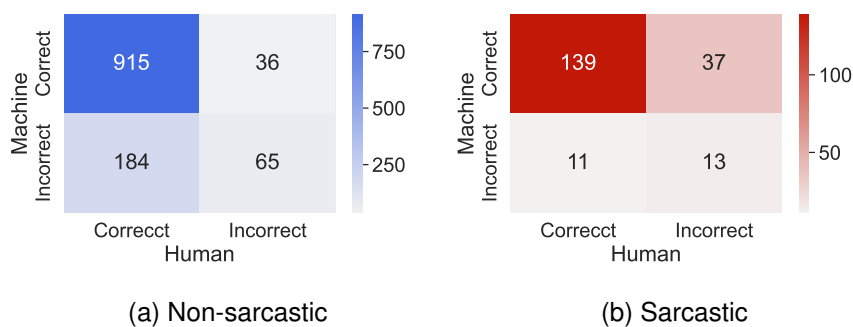


Figure 6.1: Prediction distribution for task A (Arabic).

Non-sarcastic

Table 6.3 shows some non-sarcastic examples for all the possible scenarios. When looking at the examples that were detected by both humans and machine, i.e. easy

		Human	
		Correct	Incorrect
Machine	Correct	1. لبت اكون شخص ما يفكر كثير I wish I was a person who doesn't think a lot 2. أحب لما انسال عن كل التفاصيل I like when I am asked about all the details	3. قاعدة اكل شيبس واتحسر على ايامي الضايعة اف I am eating snacks and bemoaning lost days 4. جماعه ممكن لما حد يضايقتني انا اللي از عل مش هو ؟ Would it be possible when someone annoys me that I get annoyed and not them?
	Incorrect	5. راسي رح ينفجر My head will explode 6. لا انا اليوم الضغط عندي بيرتفع No! it seems that I will have a high blood pressure today	7. تعديت مرحله البيتوتي صرت غرفوفي I moved from someone who stay at home to someone who stays in the room 8. شعره سي و يشببهه المكنسة His hair is bad and looks like a broom

Table 6.3: Arabic non-sarcastic examples.

cases, we noticed direct sentences with common words that are used frequently (Table 6.3, items 1 and 2). Also, given that Arabic has free word-order syntax, the structure of these sentences is also clear and direct, without any changes to the traditional word-order of an Arabic sentence. These examples are also quite close to modern standard Arabic (MSA), without many dialectal words or spelling variations. The examples that neither humans nor the model detected were sentences that included wordplay or some changes to the spelling of the word, an example of this is item 7 (Table 6.3). The author changed the structure of the words *غرفة* “ghurfah” (room) and *بيت* “bayt” (house) to *غرفوفي* “gharfūfi” and *بيتوتي* “baytūtī” (dialectal derivation to mean someone who love being in the room/house). This example is humorous but not sarcastic, according to the definition we adopted. The speaker is expressing that they are not outdoorsy and that they moved from walking in their house to walking in their room. Also, sentences that included sarcasm along with the non-sarcastic rephrase, were misclassified (Table 6.3, item 8). The annotators considered such cases to be sarcastic despite the annotation instructions asking them to consider sentences with indirect expressions to be sarcastic. Humans were better at detecting non-sarcastic sentences that contain descriptions that appear metaphorical, and in some cases they are, but are so commonly used that humans consider them direct descriptions (Table 6.3, items 5 and 6). The model performed better than humans for sentences that contain exaggerated complaints or descriptions. Humans assumed these cases to be sarcastic, but in reality, the meaning is expressed directly (Table 6.3, items 3 and 4). In item 3, the speaker is sad about her wasted life. However, the used way of expression implies that she is watching the sad/wasted moments of her life as someone watches a movie.

Sarcastic

Table 6.4 shows some sarcastic examples for the possible scenarios. The most commonly used and easily detected sarcastic sentences were those that contained the usage of words in uncommon contexts like item 1 (Table 6.4). In this example, the author used the word **كلور** (chlorine) in the context of addiction, which does not match the reality as humans cannot consume chlorine or be addicted to it. Proverbs and idioms (Table 6.4, items 2 and 4) were common among this set and it seems that Arabic speakers tend to rely on these meaning-dense phrases to express their feelings towards something or reply to someone. Another common pattern to express sarcasm is to use animals or objects or their attributes as adjectives to describe someone or something, e.g., item 3 (Table 6.4). On the other hand, the most challenging sarcastic examples were those that require specific context, either culturally or based on the sociocultural background and personality of the speaker (Table 6.4, items 8 and 9). Also, understanding of the dialect plays an important role here. Humans were better at detecting sarcasm that is expressed using complex metaphors that require specific world knowledge (Table 6.4, items 6 and 7). The model was better at detecting sarcasm that is expressed using dialect-specific words (Table 6.4 item 5).

		Human	
		Correct	Incorrect
Machine	Correct	1. شعب بيتعاطى كلور LT: a population addicted to chlorine IM: they are stupid 2. الي ميعرفش يقول عدس LT: this who doesn't know says lentil (proverb) IM: you/they don't know the whole story 3. قده قد الفأرة LT: he is the size of a mouse IM: He is tiny/short	4. طول البال يهد جبال LT: patience destroys mountains (Egyptian proverb) IM: patience and perseverance pay off 5. بدري!! الليش مروح هلكت ... خليك اخرى شوي LT: too early! why are you here, stay a little bit more (uses specific terms from some regions in Palestine) IM: you stayed for too long, you should leave
	Incorrect	6. عابزينا نتعلم عن بعد واحنا اصلا مش بنفهم عن قرب LT: they want us to learn remotely while we don't understand in person IM: we don't understand what we learn/bad education system 7. يستهلك المواطن الأردني ٧٩% من حسناته أثناء قيادة المركبة في كافة شوارع المملكة الحبيبة LT: the Jordanian citizen loses 79% of his good deeds while driving in the beloved Kingdom streets IM: the streets in Jordan are bad/ it is irritating to drive in Jordan	8. ابغي تعارف جاد LT: I am looking for a serious relationship IM: I want to talk to you, but I am not serious (depends on the context) 9. عد زوال هذه الأزمة إن شاء الله، ستشهد مصر اعظم موسم افراح عرفه التاريخ LT: After this struggle, Egypt will see the largest weddings season IM: depends on a specific context for the speaker.

Table 6.4: Arabic sarcastic examples. LT stands for literal translation. IM stands for intended meaning.

Pairwise Sarcasm Identification

Generally, pairwise sarcasm identification was easier than sarcasm detection. Figure 6.2 shows the quantitative difference in performance between humans and machines. There were only two sentences that neither humans nor the model detected, the first one is a Maghrebi dialect sentence (Table 6.5, item 7) and it seems that the annotators are not familiar with this dialect. For the model, the reason for this is probably the fact that Maghrebi dialect is the one with the fewest examples in iSarcasmEval’s training data. The other sentence (Table 6.5, item 8) included extremely implicit sarcasm, which can be considered present in both the sarcastic and non-sarcastic rephrases. It seems that this case caused confusion for both humans and the model. For the other cases, no clear pattern was observed.

		Human	
		Correct	Incorrect
Machine	Correct	1. S: معنديش فكه تاخذ برميل بتترول Don't you have change so you take on barrel of oil NS: سعر البترول يصل لادني مستوي Oil prices reach all time low 2. S: اجي حبيب قلبك Your beloved one has arrived NS: اجي الحدا الي بتكرهه The one you hate has arrived	3. S: يطعمك الحج والناس راجعة May you go for pilgrimage while people are coming back NS: هلا لحسيت ع حالك You just knew! (it is too late) 4. S: ابو تريكة امير القلوب Aboutrika is the prince of hearts NS: أبو تريكة كل الناس بتحبه Everybody loves Aboutrika
	Incorrect	5. S: ارحمنا يا مركز الكون Oh, center of the universe, excuse us! NS: ما جينا سيرتك أصلا We didn't even mention you! 6. S: بات مع الدجاج صبح افاقي He slept in a hen house and woke up clucking NS: تخليت على المبادئ نتعك فيساع You left your principles quickly	7. S: لوكان حراث يحرث وطنه If he was a good farmer, he would farmed his land NS: حاطر روحك قادر و نت صفر You think you are string (or good), but you are a zero 8. S: اول عقوبه يحصل عليها السعودي في مرحلة البلوغ The first punishment a Saudi gets when becoming an adult is his ID photo NS: اول عقوبه يحصل عليها السعودي عند البلوغ صورته القبيحة في البطاقة The first punishment a Saudi gets when becoming an adult is his ugly ID photo

Table 6.5: Arabic pairs of sarcastic texts and their non-sarcastic rephrases.

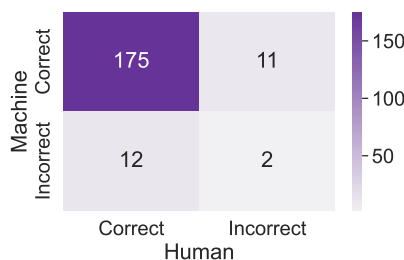


Figure 6.2: Performance of human vs machine on task C.

Thematic Error Analysis

The previous subsection provided a general overview and discussion of the nature of errors made by both humans and the machine model. To better categorise and quantify these errors, we annotated the sarcastic sentences in the test set of subtask A (sarcasm detection) according to the set of themes that we discovered in our initial error analysis. Table 6.6 provide detailed statistics of the available themes for Arabic.

Theme	N	H_{Er}	M_{Er}
Idioms	58	13	7
Proverbs	45	10	0
Referencing specific context, world knowledge	45	15	12
Complex metaphors, world knowledge	45	11	8
Dialect specific words	21	8	1
Referencing animals or objects	11	0	0
Words in uncommon context	8	0	0

Table 6.6: Sarcasm themes among sarcastic Arabic sentences. N : number of examples, H_{Er} : human error, and M_{Er} : machine error

Humans misclassified 50 sarcastic texts while the machine model misclassified 24 sarcastic texts. Most of the human errors are due to a lack of world knowledge (52%), idioms (26%) and proverbs (20%). The machine model was mostly affected by a lack of world knowledge which caused 83% of the errors.

6.1.4 Discussion

Here we revisit and answer our research questions and provide some additional discussion.

RQ5.1.1: How do humans perform on author-annotated sarcasm detection tasks? Human annotations from non-authors of the text are vastly different from the labels provided by the authors themselves. This suggests that there is an important difference between *intended* and *perceived* sarcasm, as suggested by Oprea and Magdy (2020a), who also argue that first-party annotations are more reliable as being sarcastic is an intentional act. Not only for sarcasm, but also for which intention and perception may not be consistent, the use of third-party annotations has serious implications for the reliability of our datasets' ground truth. As shown in our results, there are cases where

these annotations do not align with the labels provided by the texts' authors themselves.

RQ5.1.2: How does human performance on these tasks compare with state-of-the-art text classifiers? We found that on their own, humans performed almost as well as the state-of-the-art sarcasm detection systems, but when working together using majority voting, humans achieve the best results. On the other hand, using majority voting for the systems led to worse performance. This suggests that humans provide complementary knowledge when it comes to the task of sarcasm detection, while the text classification models' predictions typically have high overlap with one another.

RQ5.1.3: What makes sarcasm challenging for both humans and classification models? We observed that idioms and proverbs were quite common (around 23% of sarcastic sentences in the test set) and detected easily by humans and the models. It seems that Arabic speakers rely heavily on proverbs and the dense meaning they contain, while having a shared cultural context. The other common Arabic sarcasm pattern is to say "You look/act like animal/object", which is often used for derogatory remarks.

However, the most challenging cases were the sentences that require additional context. This can be in the form of conversational context, information about the author of the text, world knowledge, and dialectal awareness. For example, the model was better than humans in detecting sarcasm in sentences that used local words which some of the humans may be unfamiliar with. Given that MARBERT ([Abdul-Mageed et al., 2021a](#)) was trained on 1B tweets, it seems that it had better coverage of some specific dialects than the annotators. Interestingly, this model was able to classify Levantine examples better than the Maghrebi ones, which could be attributed to a possible bias in the training data.

The fact that missing context led to a large number of error cases implies that it is necessary for detection systems to have representations for world knowledge and cultural background, and be aware of the language/dialect of the sarcastic utterance. Progress in this direction is possible: for example, dialectal awareness can be addressed by using language models that are trained on a large dialectal variety.

6.1.5 Recommendations

Based on analyses and discussions in the previous subsections, we make the following recommendations:

- **Avoid 3rd-party annotations.** We should re-evaluate third-party annotation,

used to create ArSarcasm, as a method to create datasets of subjective content, particularly when author intention is important. The analysis of the performance of human annotators shows that their performance is comparable to state-of-the-art models. However, the performance of both the models and humans still has much room for improvement.

- **Develop models that incorporate context.** In order to improve the performance of detection models, we need to better incorporate contextual information such as cultural references, author tendencies, world knowledge, and dialect awareness. The need for this has been demonstrated through the fact that both humans and models failed to detect sarcasm that relied on such information.
- **Include contextual features in shared datasets.** Further, in order to train these models, sarcasm detection datasets that contain a wealth of contextual information should be created and released, especially conversational and author-level information which cannot be obtained from external knowledge bases. However, releasing this type of data brings new challenges in the space of privacy, as conversations contain texts written by other authors who may not have consented to sharing their content, and including more author-level information may lead to deanonymization and loss of privacy.
- **Build accurate representations of idioms and proverbs.** More focus should be put into building accurate representations of idioms and proverbs, which are extensively used in sarcastic communication.

6.2 Sarcasm and Arabic Dialects

In the previous section, we provided an in-depth discussion of humans' performance on sarcasm detection and we compared it against the performance of state-of-the-art machine learning models. In our analysis, we showed that one of the challenges for sarcasm detection is dialect awareness and familiarity. In this section, we extend our analysis to focus on how dialects can affect Arabic speakers' ability to understand and identify sarcastic content. Sarcasm understanding across dialects falls under the umbrella of dialect intelligibility among different dialects speakers. In the context of Arabic NLP, there is a lack of in-depth analyses on how Arabic dialects are understood across various regions or among speakers of different dialects. Nevertheless, how this can affect people's ability to understand sarcasm. In this section, we aim to answer the following questions:

- **RQ5.2.1:** How do speakers of different dialects perform on the sarcasm detection task?
- **RQ5.2.2:** How do speakers of different dialects understand text written in other dialects?

We answer these questions through collecting third-party annotations for SemEval’s 2022 task 6 (iSarcasmEval) dataset. We collect both sarcasm and dialect labels from third-party annotators, and we analyse the variation of performance based on annotators’ mother dialect, familiarity with other dialects, and gender. Our analysis shows that: (1) sarcasm annotations are more trustworthy if they are provided by native dialect speakers; (2) females tend to perform better than males on the sarcasm detection task; (3) annotators tend to better understand and identify their own dialect; (4) annotators are prone to confuse dialects with each other; and (5) Egyptian dialect and MSA are the easiest to identify in written text.

We also argue that our analysis, apart from showing the variation of sarcasm understanding among different Arabic dialects speakers, also shows that dialect familiarity and awareness can be detrimental to any data annotation task and that researchers should choose annotators carefully. The data used for this work with all labels are made publicly available⁴.

6.2.1 Related Work

Data Annotation and Subjectivity

Most NLP applications rely on manually annotated data. These annotations are collected from annotators from different cultures and backgrounds. Previous works acknowledged the effects of subjectivity on the quality of datasets. However, the literature lacks in-depth analyses or attempts to mitigate this issue. [Rottger et al. \(2022\)](#) tried to approach this issue through suggesting new paradigms for data annotation. They suggest that dataset creators follow either the descriptive or the prescriptive paradigm. The descriptive paradigm encourages annotator subjectivity, whereas the prescriptive paradigm discourages it. They also argue that dataset creators should explicitly aim for one or the other. For Arabic, dialect intelligibility and understanding can be one of the subjective factors affecting the data annotation process. The literature of Arabic NLP lacks in-depth analyses on the effects of dialect familiarity on the quality of data annotations or how people understand different dialects. [Habash et al. \(2008\)](#) approached

⁴<https://github.com/iabufarha/arabic-dialect-familiarity>

the dialectal variety focusing on creating standard annotation guidelines identifying dialect switching between MSA and at least one dialect. Zaidan and Callison-Burch (2014) mentioned that annotators tend to over-identify their dialect. We add to this line of work by exploring how annotators understand different dialects. We also analyse the quality of their labels on one of the most subjective tasks, sarcasm detection.

6.2.2 Methodology

In this section, we describe our methodology for the analysis of dialects comprehension during data annotation tasks, using sarcasm detection as a case study. We initially discuss the dataset we used and its ground-truth labels. Then we explain collecting third-party labels from annotators of different dialects, which will be compared later to the ground-truth labels for the analysis process.

Dataset

Similar to the analysis presented in Section 6.1, we use SemEval-2022 Task 6, iSarcasmEval, datasets. For this work, we use the test set of Arabic subtask A (sarcasm detection). The test set consists of 1400 sentences, 200 of which are sarcastic and 1200 non-sarcastic. Each of the sentences has two labels provided by the author of the sentence: the dialect of the sentence (out of five dialects) and whether the sentence is meant to be sarcastic or not. Table 6.7 shows the statistics over the available dialects.

Dialect	Total	Sarcastic	Non-sarcastic
Egyptian/Nile	520	131	389
MSA	482	16	466
Gulf	176	10	166
Levantine	168	22	146
Maghrebi	54	21	33

Table 6.7: Distribution of the Arabic iSarcasmEval test set over the dialects.

Third-party Annotations

To analyse the performance of speakers of different dialects, we collected third-party annotations using Appen⁵ platform. For each sentence, we collected *five annotations*.

⁵<https://appen.com>

We allowed only native Arabic speakers to participate. Before starting the annotation process, each annotator is presented with test questions and only those who answer all the questions correctly would be allowed to participate in the annotation process. The test questions were sampled from a set of sentences that are clearly sarcastic/non-sarcastic. We used this approach to make sure that the annotators are not giving random answers and to avoid introducing any bias before the annotation. For each sentence, we asked annotators to provide the following:

- Sarcasm label indicating whether the text is sarcastic or not.
- Dialect label out of five: MSA, Egyptian (Nile), Gulf, Levantine, and Maghrebi.
- Mother dialect, which is the dialect the annotator grew up speaking.
- Known dialects, which are the dialects the annotator is familiar with.
- Gender of the annotator (either male or female).

A total of 22 annotators participated in our survey, 15 males and 7 females. Table 6.8 provides the distribution of the annotators according to their mother dialect and the dialects they are familiar with.

Dialect	Mother dialect	Known by
Egyptian/Nile	11	21
Levantine	6	10
Gulf	1	7
Maghrebi	4	5
MSA	-	16

Table 6.8: Annotators' details. The table shows the number of annotators who speak a specific dialect as a mother tongue and the number of annotators who mentioned that they know a specific dialect.

6.2.3 Results and Analysis

Sarcasm Detection

We examined the annotators' performance on the subjective task of sarcasm detection, which requires annotators to be able to understand the text to provide correct labels and is found to be a highly challenging task for annotators in different languages. Table 6.9 shows the annotators' performance on sarcasm detection. From the table, Levantine speakers seem to perform better on this task, followed by Gulf speakers. In order to

have a better understanding, we analyse the performance over each dialect. Figure 6.3 shows the performance of speakers of a specific dialect on all the dialects. The figure shows $F1^{\text{sarcastic}}$ score and the number of annotations for the respective dialect. The results show that speakers of the Egyptian (Nile) dialect struggle to detect sarcasm written in MSA. Also, speakers of Maghrebi and Egyptian dialects struggle to identify sarcasm expressed using the Gulf’s dialect. The results show that Levantine and Gulf speakers perform relatively well on all the dialects. Generally, the annotators achieved the highest score when the text was in Egyptian or their mother dialect.

Speaker’s dialect	$F_1^{\text{sarcastic}}$
Egyptian/Nile	0.50
Gulf	0.53
Levantine	0.58
Maghrebi	0.48

Table 6.9: Sarcasm detection performance ($F_1^{\text{sarcastic}}$) of speakers of different dialects.

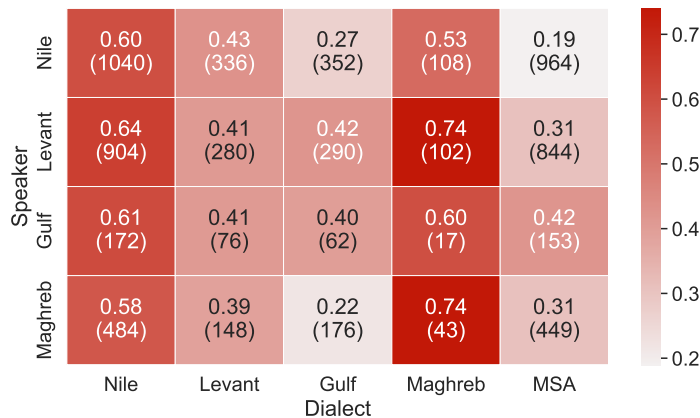


Figure 6.3: Sarcasm detection performance ($F_1^{\text{sarcastic}}$) of different dialects speakers on each dialect. Original dialect labels were used. Annotation counts are indicated in brackets.

Sarcasm Detection - Dialect Familiarity

Figures 6.4 and 6.5 show the performance of annotators in two cases: when the text’s dialect is one that they are familiar with and when it is not. When considering the case when the text’s dialect is one that the annotators are familiar with (Figure 6.4), the

annotators have the highest performance on the Egyptian (Nile) dialect. These scores indicate that the annotators are truly familiar with the Egyptian (Nile) dialect. When looking at the cases where people are unfamiliar with the dialect, the performance is inconsistent. For example, the performance of Maghrebi speakers on texts in Levantine is higher for annotators who indicated that they are not familiar with the Levantine dialect. Another example is Levantine speakers' performance on Maghrebi texts. Such inconsistencies indicate that some annotators might have provided a guess regarding the sarcasm label or that they underestimated their familiarity with the respective dialect.

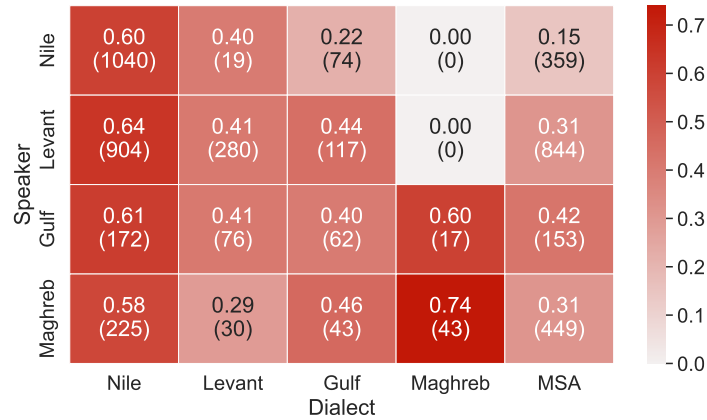


Figure 6.4: Annotators' sarcasm detection performance ($F_1^{\text{sarcastic}}$) when the dialect is *known* to them. Annotation counts are indicated in brackets.

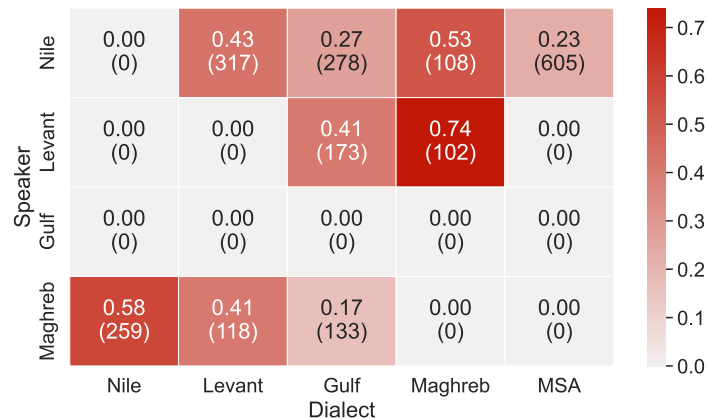


Figure 6.5: Annotators' sarcasm detection performance ($F_1^{\text{sarcastic}}$) when the dialect is *unknown* to them. Annotation counts are indicated in brackets.

Figures 6.6 and 6.7 show the performance when the annotators identified the dialects either correctly or incorrectly. The figures show that the performance is generally higher when the annotators identify the dialect correctly. This goes along with the previous observation that the annotators performed better on dialects they are familiar with. The exceptions are the performance of Levantine speakers on Maghrebi dialect, Maghrebi speakers on Levantine, and Nile speakers on Gulf dialect. Levantine speakers performed slightly better on MSA when they incorrectly identified the dialect. This goes along with the previous observation that, indeed, some annotators might be guessing the labels.

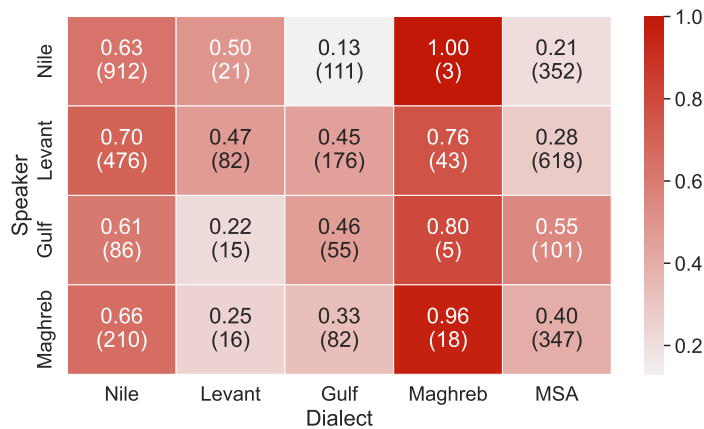


Figure 6.6: Annotators' sarcasm detection performance ($F_1^{\text{sarcastic}}$) when they *correctly* identified the dialect. Annotation counts are indicated in brackets

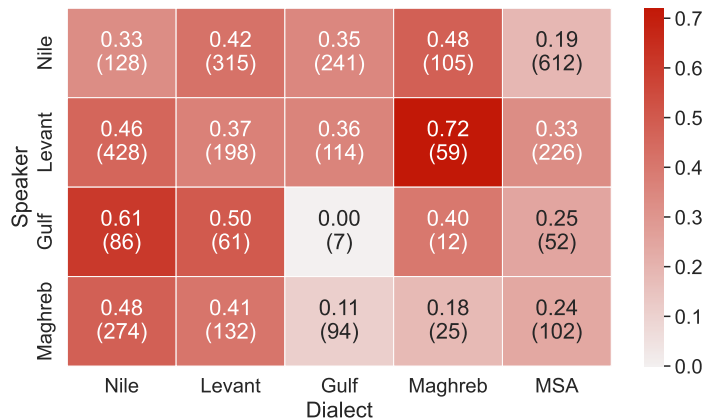


Figure 6.7: Annotators' sarcasm detection performance ($F_1^{\text{sarcastic}}$) when they *incorrectly* identified the dialect. Annotation counts are indicated in brackets.

Sarcasm and Gender

We further analysed the performance of annotators based on their gender. Figure 6.8 shows the performance over dialects based on the annotators' gender. From the figure, it is noticeable that females perform better than males at detecting sarcasm. Females performed better than males on all dialects except MSA where the performance is quite comparable. This observation matches the one found by [Oprea and Magdy \(2020b\)](#).

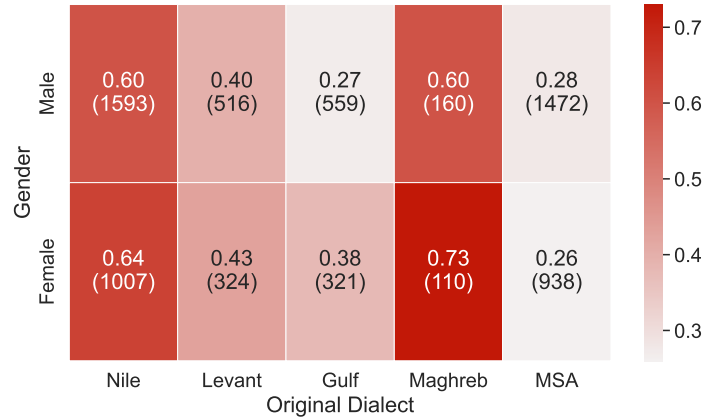


Figure 6.8: Sarcasm detection performance ($F_1^{\text{sarcasmic}}$) based on the annotators' gender. Annotation counts are indicated in brackets.

Dialect Identification

Figure 6.9 shows the accuracy of annotators in identifying the dialects. From the figure, it is clear the annotators, except Egyptian speakers, were able to identify MSA. Egyptian and Gulf speakers performed best on their dialect. Levantine and Maghrebi speakers performed better on dialects other than their own. Figure 6.10 shows the distribution of assigned dialect labels compared to the original ones. The results show that Egyptian and MSA are the easiest to identify. However, the annotators seem to confuse other dialects, especially Levantine and Maghrebi. Figures 6.11, 6.12, 6.13, and 6.14 provide a clearer picture of how speakers of one dialect identified other dialects. As shown in Figures 6.11 and 6.13, Egyptian and Gulf speakers excel at identifying texts in their dialect. Figure 6.14 shows that Maghrebi speakers seem to confuse their dialect with MSA. Levantine speakers (Figure 6.12) seem to confuse their dialect with the Gulf dialect. Similar to Figure 6.10, most annotators tend to easily identify MSA, except for Egyptian speakers who confuse it for Egyptian dialect. Gulf speakers seem to confuse Levantine and Maghrebi for the Gulf dialect.

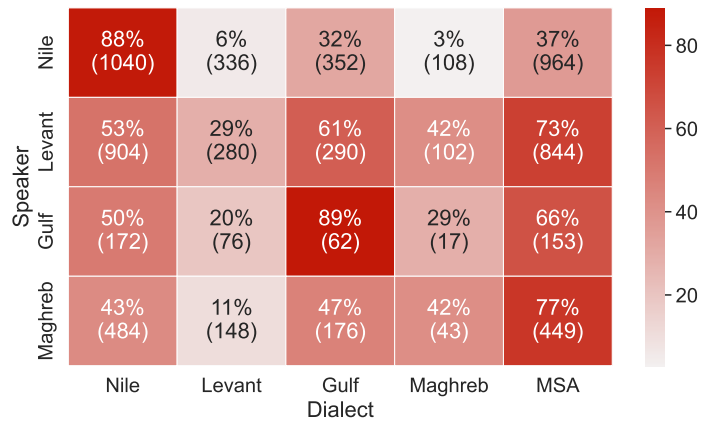


Figure 6.9: Dialect identification accuracy of annotators speaking different dialects. Annotation counts are indicated in brackets.

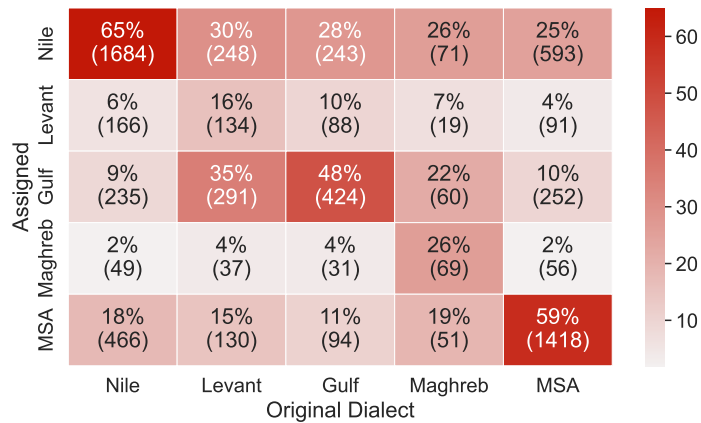


Figure 6.10: Assigned dialect labels vs the original ones. Annotation counts are indicated in brackets.

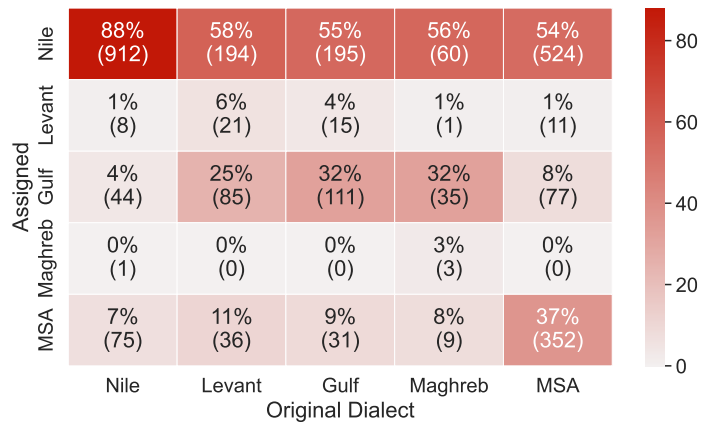


Figure 6.11: Egyptian (Nile) speakers' dialect identification performance.

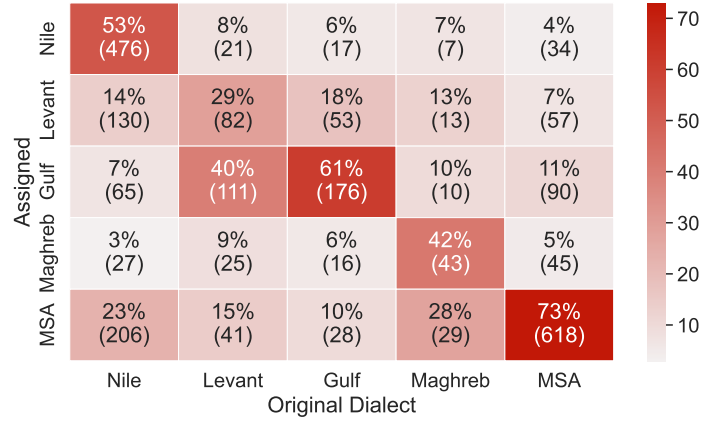


Figure 6.12: Levantine speakers' dialect identification performance.

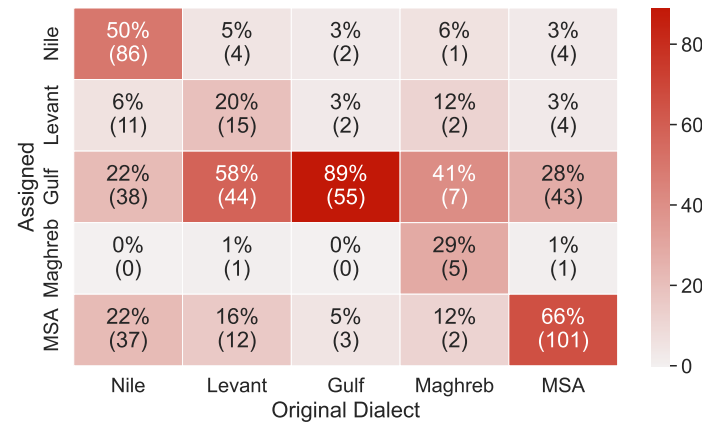


Figure 6.13: Gulf speakers' dialect identification performance.

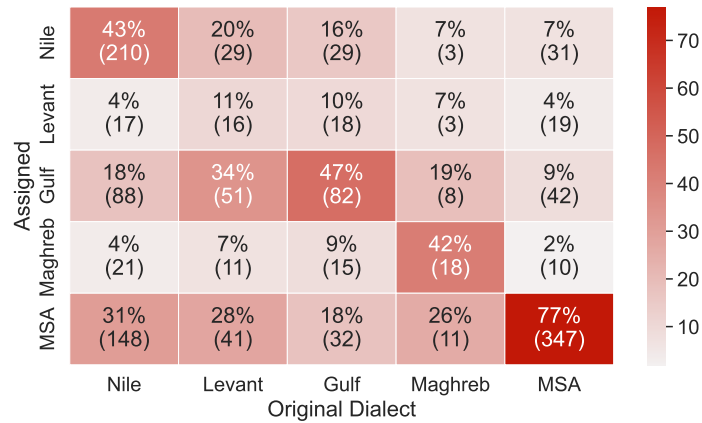


Figure 6.14: Maghrebi speakers' dialect identification performance.

6.2.4 Discussion

In this section, we provide a discussion of the results mentioned in Section 6.2.3. We also revisit and answer our research questions as follows:

RQ5.2.1: How do speakers of different dialects perform on the sarcasm detection task? As discussed in Section 6.2.3, annotators tend to better understand sarcasm expressed in their dialect. This is due to the fact that annotators unfamiliar with a dialect would struggle to grasp the complete meaning of a sentence. Also, the fact that sarcasm usually relies on cultural references that can be specific to a region/dialect means that people unfamiliar with the dialect would not be able to understand such references. This observation aligns with the findings in (Oprea and Magdy, 2020b), where the authors found that English language nativeness and country are significantly influential on sarcasm understanding. Indeed, these factors should be considered when collecting third-party annotations for Arabic data. Although there are many shared linguistic and cultural aspects among Arabic speakers, there are still some local differences. Those are embodied in culture, traditions, and dialects. Thus, it is necessary to have native speakers who are aware and familiar with these differences, annotating subjective and linguistically complex data like sarcasm.

RQ5.2.2: How do speakers of different dialects understand other dialects? There are some similarities between dialects and, to some extent, people speaking different dialects can understand each other. However, as shown in Section 6.2.3, annotators tend to confuse some dialects for different ones. For example, Egyptian speakers tend to over-identify their own dialect, assuming that more than 50% of other dialects to be Egyptian. This observation is similar to the behaviour observed in (Zaidan and Callison-Burch, 2014). Similar behaviour is observed with Gulf speakers towards Levantine. Such over-identification behaviour, and given the large number of Egyptian annotators, might introduce bias into datasets. Egyptian, Gulf, and Maghrebi speakers tend to perform better on their dialect. Levantine speakers' performance was inconsistent and they seemed to confuse Levantine for Gulf. This could be due to the spectrum of variation within the Levant countries from north to south, where the southern Levantine dialect is closer to the Gulf dialect.

The confusion between the dialects might be due to the fact that these dialects share many words or the differences are mostly phonological. Also, due to the slight differences between dialects' orthography, annotators might confuse sentences in dialects they are unfamiliar with and assign them to a different one. This phenomenon is

clear in Section 6.2.3, where Levantine speakers had better performance on MSA for sarcasm detection, but they assigned an incorrect dialect label.

In our analysis, we also included the performance of annotators based on gender. Based on the results in Section 6.2.3, female annotators seem to detect sarcasm better than male annotators. With the small number of annotators and the available data, we cannot provide an explanation for this observation. Future works should consider studying this in a better-designed setup that considers other factors such as educational background and personality traits.

6.3 Summary

This chapter provides an in-depth analysis of sarcasm detection on the proposed iSarcasmEval dataset. To answer the research question “*RQ5: How do Arabic speakers perform on sarcasm detection and how does dialectal variation affect that?*”, we collected third-party labels from annotators and analysed their performance on sarcasm detection and compared it to state-of-the-art detection models. Our analyses show that sarcasm detection is challenging for humans. We also include a thorough error analysis, revealing that the most challenging sarcastic sentences are those that require additional contextual information to resolve accurately, suggesting that future work focuses on creating context-rich datasets and models with the ability to adequately leverage contextual information. Our analyses show that idioms and proverbs are common linguistic tools used to express Arabic sarcasm, yet trained models often struggle with examples that contain them. We also analyse humans’ performance on sarcasm detection on the dialect level. Our analysis shows that the performance of annotators varies based on the annotators’ familiarity with the text’s dialect. Generally, the performance was higher when annotators were familiar with the dialect. Based on the analysis, it is clear that dialect familiarity affects how humans understand sarcasm.

Chapter 7

Conclusions & Future Work

7.1 Conclusions

In this thesis, we focused on Arabic sarcasm detection. The literature survey in Chapter 2, showed that there is a gap that needs to be filled in the research on Arabic sarcasm detection. There are a few works on the topic and at the time of starting this work, there were no publicly available datasets. In our work, we tried to fill in the research gaps from different angles. First, due to a lack of publicly available datasets, we tried to create our own Arabic sarcasm dataset. Over the course of this thesis, we explored various methods to collect sarcasm datasets, the best of which is to ask people to label their words, i.e. intended sarcasm. Second, we explored and benchmarked the performance of state-of-the-art models, which showed a low performance, emphasising the challenging nature of sarcasm. Finally, we conducted a thorough analysis comparing humans' against machine learning models' performance on the sarcasm detection task. We followed this with an analysis showing that Arabic dialect familiarity has a detrimental impact on how people understand sarcasm or any other dialectal text.

In **Chapter 3**, to answer **RQ3**, we discussed our effort to create an Arabic sarcasm dataset. We started with the re-annotation of existing Arabic sentiment analysis datasets. The annotation included sarcasm, sentiment and dialect labels. The analysis showed that sarcasm is present in Arabic sentiment analysis datasets, with around 16% of the texts in the used datasets being sarcastic, which answers **RQ1**. Also, our analysis shows the highly subjective nature of such datasets, which was demonstrated by the major change in sentiment labels. The experiments show the gap between sentiment analysis systems' performance on non-sarcastic tweets compared to sarcastic tweets, which answers **RQ2**.

In **Chapter 4**, in our effort to create an Arabic sarcasm dataset that has trustworthy

labels, we decided to create a first-party annotated dataset. We created our new dataset through asking people to provide sarcastic sentences on the fly. We believe that this approach mitigates the fallbacks of other approaches and provides the most reliable labels, which answers **RQ3**.

In **Chapters 3** and **4**, we provided the details of the shared tasks that we organised. These shared tasks helped bring attention to the Arabic sarcasm detection task. Also, they helped enrich the literature with a plethora of models for sarcasm detection through the submissions of the participating teams. The analysis of the submissions along with the experiments in **Chapter 5** provides the answer to **RQ4**.

Finally, in **Chapter 6**, to answer **RQ5**, we provide our in-depth analyses of sarcasm detection. We compare humans' performance against state-of-the-art models. Our analyses show that sarcasm detection is challenging for humans with performance only slightly better than trained models, and only when using majority voting between the human predictions. The low human performance emphasises the subjective nature of sarcasm and indicates that third-party labels for subjective tasks are noisy. Consequently, we urge the community to re-evaluate third-party annotations for extremely subjective tasks, such as sarcasm, and use first-party labels. We extend the analysis to focus on Arabic dialects, where we compared the performance on sarcasm detection in cases when the annotators were familiar or unfamiliar with the text's dialect. The analysis shows that dialect familiarity affects how annotators understand a given text and, consequently, their understanding of the expressed sarcasm.

To summarise, we achieved the following contributions:

- Creating and sharing the first publicly available Arabic sarcasm dataset, ArSarcasm. The details of ArSarcasm are provided in Chapter 3.
- Creating and publicly sharing three datasets. In our effort to find the best way to create an Arabic sarcasm dataset, we created three datasets: ArSarcasm, ArSarcasm-v2, and iSarcasmEval. The details of these datasets are provided in Chapters 3 and 4.
- Bringing attention to the Arabic sarcasm detection task. This was achieved through the organisation of two shared tasks, which also helped enrich the literature with various models and approaches for sarcasm detection.
- Benchmarking state-of-the-art models on sarcasm detection. This was achieved through the experiments in Chapter 5 and the details of the teams participating in the shared tasks.
- Providing an in-depth error analysis of humans' performance on sarcasm detec-

tion. This analysis highlights the challenging cases in sarcasm detection and provides an anchor for future research to improve sarcasm detection models.

- Analysing the effect of dialectal variation on the process of data annotation, where we show that dialect familiarity affects how annotators understand texts in a specific dialect, hence their labels when annotating data.

7.2 Implications

We believe that our effort in this work has helped in enriching the Arabic NLP literature. However, we barely scratched the surface, and there are still many open questions. First of all, sarcasm detection is still a challenging task, and it would require more work to improve performance. In our work, we showed some of the patterns and features that make sarcasm challenging, such as idioms and proverbs. The main challenge in sarcastic language and figurative language, in general, is the subtle implicitness. In the context of Arabic, this is magnified due to the large dialectal variation. Those dialects do not just represent differences in the language but also include cultural and historical differences. Modelling these variations is challenging, even for most advanced models such as ChatGPT¹. Appendix D provides examples of our experiments on using ChatGPT for sarcasm detection using some examples from iSarcasmEval dataset. The answers provided by the model show that sarcasm is challenging and that it is extremely challenging to resolve cultural references. We believe that future research should focus more on representing proverbs and idioms, which would help computational models understand their meaning. Also, these efforts need to consider the dialectal variation of these proverbs across countries. Additionally, more work should focus on sarcasm detection and explanation as the detection itself might be grounded in inaccurate/incorrect explanations.

7.3 Limitations & Future Directions

Up to the start of this work, there were a few works on Arabic sarcasm detection. In our work, we tried to fill the research gap through creating resources, organising shared tasks, and providing in-depth analyses. In our work, we created three datasets that were used to organise two shared tasks, which helped bring attention to the Arabic sarcasm detection task. We also provided experiments and analyses using state-of-the-art models. Finally, we provided error analyses discussing what makes sarcasm challenging for humans and machine models. In our work, a major limitation was

¹<https://openai.com/blog/chatgpt>

access to Arabic content and Arabic annotators. This has limited our work in terms of dataset creation and in terms of the analyses we provide focusing on Arabic dialects. Additionally, although we tried to provide extensive experiments on detection models, our experiments were limited to large language models, without an extensive deliberate effort to improve their performance. Finally, our analyses were limited by the available data and the access to annotators, which might have introduced some biases, especially when analysing the effect of dialect familiarity on sarcasm detection. Future research directions include:

- **Creating new larger datasets, with better dialectal coverage:** In our work, we created three datasets, but those were limited by the access to the data and annotators. In our work, we argue that first-party sarcasm labels are more reliable than third-party ones. However, this data collection approach is cost and time expensive. The collected data was relatively small and lacked a balanced dialectal variation. Future research should consider creating larger datasets, with better dialectal coverage.
- **Creating context-rich datasets:** Our analysis in Chapter 6 shows that one of the major challenges to detecting sarcasm is the lack of contextual information. Future research should explore how to enrich sarcasm datasets with such information.
- **Exploring and improving sarcasm detection models:** Future research should try to explore and improve sarcasm detection models. Such research can benefit from the datasets and analyses provided in this thesis. A starting point can be the findings and discussions provided in Chapter 6 on what makes sarcasm challenging. For example, researchers can explore how to embed contextual information and world knowledge into models. Additionally, newer, more advanced models can be used.
- **More analytical studies:** In our analyses in Chapter 6, we only compared humans' performance against the top teams in iSarcasmEval shared task. Future research can consider comparing the performance against more advanced models. Also, when analysing the effect of dialect familiarity, a major limitation was the dialect distribution of third-party annotators. Future work can consider a larger pool of annotators with better dialectal coverage. Future studies can also consider more sociocultural variables for the analysis.
- **Sarcasm detection for downstream tasks:** In this work, we showed that sarcasm can be disruptive for sentiment analysers. Future research can explore how to utilise sarcasm detection models to improve the performance on sentiment analysis. Since sarcastic language can be used to express hate-speech, racism, or bullying, researchers can explore how to utilise sarcasm to improve models targeting these tasks.

- **Sarcasm and speech:** In this work, we only considered sarcasm in written text. Future research can focus on detecting sarcasm in speech and utilise the additional speech information. Also, work on sarcasm detection can benefit text-to-speech systems, which would enable such systems to identify sarcasm and produce proper sarcastic intonation.
- **Multimodal sarcasm detection:** In this work, we only considered sarcasm in textual form. Future research can focus on multimodal sarcasm detection and analysis. For example, research can focus on analysing memes and the accompanying quotes. Also, a mixture of speech and visual signals can be used to identify sarcasm in videos.
- **Sarcasm and language choice:** In Chapter 3, we noticed that most of the sarcastic sentences are in dialectal Arabic (DA) and that MSA had the lowest sarcasm ratio. Hence, the sarcasm detection task also includes a latent dialect identification task. Future works can explore how people choose between DA and MSA to express sarcasm and what factors affect this choice.

7.4 Impact

At the start of this work, there were a few works on Arabic sarcasm detection. There were no publicly available datasets, models, or analyses. Over the course of this project, we tried to fill the research gaps and bring more attention to the Arabic sarcasm detection task. We organised two shared tasks, one of them is part of SemEval. The datasets and analyses we provided are the basis for the current research on Arabic sarcasm. The success of our effort is manifested in a large number of participants in our shared tasks. In SemEval, our task was the second largest shared task, with participation from 60 unique teams. Also, this was manifested by the relatively large number of citations² our work received, which is shown below:

- ArSarcasm dataset (Abu Farha and Magdy, 2020), which contains 10,547 tweets, received 82 citations.
- ArSarcasm-v2 dataset (Abu Farha et al., 2021), which contains 15,548 tweets, received 48 citations.
- iSarcasmEval datasets (Abu Farha et al., 2022a), which contains 5,503 sentences, received 50 citations.
- Sarcasm detection benchmarking paper (Abu Farha and Magdy, 2021) received 43 citations.

²As reported on Google Scholar in August 2023.

Bibliography

- Ahmed Abbasi, Hsinchun Chen, and Arab Salem. 2008. Sentiment analysis in multiple languages: Feature selection for opinion classification in web forums. *ACM Transactions on Information Systems (TOIS)*, 26(3):1–34.
- Ines Abbes, Wajdi Zaghouani, Omaima El-Hardlo, and Faten Ashour. 2020. [DAICT: A dialectal Arabic irony corpus extracted from Twitter](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6265–6271, Marseille, France. European Language Resources Association.
- Reem Abdel-Salam. 2021. [WANLP 2021 shared-task: Towards irony and sentiment detection in Arabic tweets using multi-headed-LSTM-CNN-GRU and MaRBERT](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 306–311, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Reem Abdel-Salam. 2022. [reamtchka at SemEval-2022 task 6: Investigating the effect of different loss functions for sarcasm detection for unbalanced datasets](#). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 896–906, Seattle, United States. Association for Computational Linguistics.
- Ahmed Abdelali, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. 2016. [Farasa: A fast and furious segmenter for Arabic](#). In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 11–16, San Diego, California. Association for Computational Linguistics.
- Muhammad Abdul-Mageed. 2017. [Not all segments are created equal: Syntactically motivated sentiment analysis in lexical space](#). In *Proceedings of the Third Arabic Natural Language Processing Workshop*, pages 147–156, Valencia, Spain. Association for Computational Linguistics.

- Muhammad Abdul-Mageed. 2019. Modeling arabic subjectivity and sentiment in lexical space. *Information Processing & Management*, 56(2):291–307.
- Muhammad Abdul-Mageed and Mona Diab. 2012. [AWATIF: A multi-genre corpus for Modern Standard Arabic subjectivity and sentiment analysis](#). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 3907–3914, Istanbul, Turkey. European Language Resources Association (ELRA).
- Muhammad Abdul-Mageed, Mona Diab, and Mohammed Korayem. 2011. [Subjectivity and sentiment analysis of Modern Standard Arabic](#). In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 587–591, Portland, Oregon, USA. Association for Computational Linguistics.
- Muhammad Abdul-Mageed, Mona Diab, and Sandra Kübler. 2014. [Samar: Subjectivity and sentiment analysis for arabic social media](#). *Computer Speech & Language*, 28(1):20–37.
- Muhammad Abdul-Mageed, AbdelRahim Elmadany, and El Moatez Billah Nagoudi. 2021a. [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)*, pages 7088–7105, Online. Association for Computational Linguistics.
- Muhammad Abdul-Mageed, Sandra Kuebler, and Mona Diab. 2012. [SAMAR: A system for subjectivity and sentiment analysis of Arabic social media](#). In *Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis*, pages 19–28, Jeju, Korea. Association for Computational Linguistics.
- Muhammad Abdul-Mageed, Chiyu Zhang, Houda Bouamor, and Nizar Habash. 2020. [NADI 2020: The first nuanced Arabic dialect identification shared task](#). In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 97–110, Barcelona, Spain (Online). Association for Computational Linguistics.
- Muhammad Abdul-Mageed, Chiyu Zhang, AbdelRahim Elmadany, Houda Bouamor, and Nizar Habash. 2021b. [NADI 2021: The second nuanced Arabic dialect identification shared task](#). In *Proceedings of the Sixth Arabic Natural Language Processing*

- Workshop*, pages 244–259, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- N. A. Abdulla, N. A. Ahmed, M. A. Shehab, and M. Al-Ayyoub. 2013. [Arabic sentiment analysis: Lexicon-based and corpus-based](#). In *2013 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, pages 1–6.
- Gavin Abercrombie and Dirk Hovy. 2016. [Putting sarcasm detection into context: The effects of class imbalance and manual labelling on supervised machine classification of Twitter conversations](#). In *Proceedings of the ACL 2016 Student Research Workshop*, pages 107–113, Berlin, Germany. Association for Computational Linguistics.
- Ibrahim Abu Farha and Walid Magdy. 2019. [Mazajak: An online Arabic sentiment analyser](#). In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 192–198, Florence, Italy. Association for Computational Linguistics.
- Ibrahim Abu Farha and Walid Magdy. 2020. [From Arabic sentiment analysis to sarcasm detection: The ArSarcasm dataset](#). In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 32–39, Marseille, France. European Language Resource Association.
- Ibrahim Abu Farha and Walid Magdy. 2021. [Benchmarking transformer-based language models for Arabic sentiment and sarcasm detection](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 21–31, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Ibrahim Abu Farha and Walid Magdy. 2021. [A comparative study of effective approaches for arabic sentiment analysis](#). *Information Processing & Management*, 58(2):102438.
- Ibrahim Abu Farha and Walid Magdy. 2022. [The effect of Arabic dialect familiarity on data annotation](#). In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, pages 399–408, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.
- Ibrahim Abu Farha, Silviu Vlad Oprea, Steven Wilson, and Walid Magdy. 2022a. [SemEval-2022 task 6: iSarcasmEval, intended sarcasm detection in English and](#)

- [Arabic](#). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 802–814, Seattle, United States. Association for Computational Linguistics.
- Ibrahim Abu Farha, Steven Wilson, Silviu Oprea, and Walid Magdy. 2022b. [Sarcasm detection is way too easy! an empirical comparison of human and machine sarcasm detection](#). In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5284–5295, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Ibrahim Abu Farha, Wajdi Zaghrouani, 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*, pages 296–305, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Abeer Abuzayed and Hend Al-Khalifa. 2021. [Sarcasm and sentiment detection in Arabic tweets using BERT-based models and data augmentation](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 312–317, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Mahmoud Al-Ayyoub, Safa Bani Essa, and Izzat Alsmadi. 2015. Lexicon-based sentiment analysis of arabic tweets. *International Journal of Social Network Mining*, 2(2):101–114.
- Ahmad Al Sallab, Hazem Hajj, Gilbert Badaro, Ramy Baly, Wassim El Hajj, and Khaled Bashir Shaban. 2015. [Deep learning models for sentiment analysis in Arabic](#). In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 9–17, Beijing, China. Association for Computational Linguistics.
- Mohammad Al-Smadi, Omar Qawasmeh, Mahmoud Al-Ayyoub, Yaser Jararweh, and Brij Gupta. 2018. Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of arabic hotels’ reviews. *Journal of Computational Science*, 27:386–393.
- Mohammad Al-Smadi, Bashar Talafha, Mahmoud Al-Ayyoub, and Yaser Jararweh. 2019. Using long short-term memory deep neural networks for aspect-based sentiment analysis of Arabic reviews. *International Journal of Machine Learning and Cybernetics*, 10(8):2163–2175.

- N. Al-Twairesh and H. Al-Negheimish. 2019. Surface and deep features ensemble for sentiment analysis of arabic tweets. *IEEE Access*, 7:84122–84131.
- Hamza Alami, Abdessamad Benlahbib, and Ahmed Alami. 2022. [High tech team at SemEval-2022 task 6: Intended sarcasm detection for Arabic texts](#). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 840–843, Seattle, United States. Association for Computational Linguistics.
- Abdulaziz M. Alayba, Vasile Palade, Matthew England, and Rahat Iqbal. 2017. [Arabic language sentiment analysis on health services](#). In *2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*, pages 114–118.
- Abdulaziz M. Alayba, Vasile Palade, Matthew England, and Rahat Iqbal. 2018. A combined cnn and lstm model for arabic sentiment analysis. In *Machine Learning and Knowledge Extraction*, pages 179–191, Cham. Springer International Publishing.
- Abdullah I. Alharbi and Mark Lee. 2021. [Multi-task learning using a combination of contextualised and static word embeddings for Arabic sarcasm detection and sentiment analysis](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 318–322, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Sarah O Alhumoud, Mawaheb I Altuwaijri, Tarfa M Albuhairi, and Wejdan M Alohaideb. 2015. Survey on arabic sentiment analysis in twitter. *International Science Index*, 9(1):364–368.
- Hassan Alhuzali, Muhammad Abdul-Mageed, and Lyle Ungar. 2018. [Enabling deep learning of emotion with first-person seed expressions](#). In *Proceedings of the Second Workshop on Computational Modeling of People’s Opinions, Personality, and Emotions in Social Media*, pages 25–35, New Orleans, Louisiana, USA. Association for Computational Linguistics.
- A. Aziz Altowayan and Lixin Tao. 2016. [Word embeddings for arabic sentiment analysis](#). In *2016 IEEE International Conference on Big Data (Big Data)*, pages 3820–3825.
- Mohamed Aly and Amir Atiya. 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)*, pages 494–498, Sofia, Bulgaria. Association for Computational Linguistics.

- Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mário J. Silva. 2016. [Modelling context with user embeddings for sarcasm detection in social media](#). In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 167–177, Berlin, Germany. Association for Computational Linguistics.
- Wissam Antoun, Fady Baly, and Hazem Hajj. 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*, pages 9–15, Marseille, France. European Language Resource Association.
- Wissam Antoun, Fady Baly, and Hazem Hajj. 2021a. [AraELECTRA: Pre-training text discriminators for Arabic language understanding](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 191–195, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Wissam Antoun, Fady Baly, and Hazem Hajj. 2021b. [AraGPT2: Pre-trained transformer for Arabic language generation](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 196–207, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- David Bamman and Noah A Smith. 2015. Contextualized sarcasm detection on twitter. In *Ninth International AAAI Conference on Web and Social Media*.
- Francesco Barbieri, Francesco Ronzano, and Horacio Saggion. 2014a. Italian irony detection in twitter: a first approach. In *The First Italian Conference on Computational Linguistics CLiC-it*, volume 28.
- Francesco Barbieri, Horacio Saggion, and Francesco Ronzano. 2014b. [Modelling sarcasm in Twitter, a novel approach](#). In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 50–58, Baltimore, Maryland. Association for Computational Linguistics.
- Arup Baruah, Kaushik Das, Ferdous Barbhuiya, and Kuntal Dey. 2020. [Context-aware sarcasm detection using BERT](#). In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 83–87, Online. Association for Computational Linguistics.

- Laila Bashmal and Daliyah AlZeer. 2021. [ArSarcasm shared task: An ensemble BERT model for SarcasmDetection in Arabic tweets](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 323–328, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Christos Baziotis, Prashant Mathur, and Eva Hasler. 2023. [Automatic evaluation and analysis of idioms in neural machine translation](#). In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3682–3700, Dubrovnik, Croatia. Association for Computational Linguistics.
- Kenneth R Beesley. 1998. Arabic morphological analysis on the internet. In *Proceedings of the 6th International Conference and Exhibition on Multi-lingual Computing*. Citeseer.
- Farah Benamara, Cyril Grouin, Jihen Karoui, Véronique Moriceau, and Isabelle Robba. 2017. Analyse d’opinion et langage figuratif dans des tweets: présentation et résultats du défi fouille de textes deft2017.
- Santosh Kumar Bharti, Korra Sathya Babu, and Sanjay Kumar Jena. 2017. Harnessing online news for sarcasm detection in hindi tweets. In *Pattern Recognition and Machine Intelligence*, pages 679–686, Cham. Springer International Publishing.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. [Enriching word vectors with subword information](#). *Transactions of the Association for Computational Linguistics*, 5:135–146.
- Houda Bouamor, Sabit Hassan, and Nizar Habash. 2019. [The MADAR shared task on Arabic fine-grained dialect identification](#). In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 199–207, Florence, Italy. Association for Computational Linguistics.
- Mondher Bouazizi and Tomoaki Otsuki Ohtsuki. 2016. A pattern-based approach for sarcasm detection on twitter. *IEEE Access*, 4:5477–5488.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark,

- Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Elisabeth Camp. 2012. Sarcasm, pretense, and the semantics/pragmatics distinction. *Noûs*, 46(4):587–634.
- John D Campbell and Albert N Katz. 2012. Are there necessary conditions for inducing a sense of sarcastic irony? *Discourse Processes*, 49(6):459–480.
- Marine Carpuat and Mona Diab. 2010. [Task-based evaluation of multiword expressions: a pilot study in statistical machine translation](#). In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 242–245, Los Angeles, California. Association for Computational Linguistics.
- Basilis Charalampakis, Dimitris Spathis, Elias Kouslis, and Katia Kermanidis. 2016. [A comparison between semi-supervised and supervised text mining techniques on detecting irony in greek political tweets](#). *Engineering Applications of Artificial Intelligence*, 51:50 – 57. Mining the Humanities: Technologies and Applications.
- Shammur Absar Chowdhury, Ahmed Abdelali, Kareem Darwish, Jung Soon-Gyo, Joni Salminen, and Bernard J. Jansen. 2020. [Improving Arabic text categorization using transformer training diversification](#). In *Proceedings of the Fifth Arabic Natural Language Processing Workshop*, pages 226–236, Barcelona, Spain (Online). Association for Computational Linguistics.
- Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. [ELECTRA: Pre-training text encoders as discriminators rather than generators](#). In *ICLR*.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

- Abdelghani Dahou, Shengwu Xiong, Junwei Zhou, Mohamed Houcine Haddoud, and Pengfei Duan. 2016. [Word embeddings and convolutional neural network for Arabic sentiment classification](#). In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2418–2427, Osaka, Japan. The COLING 2016 Organizing Committee.
- Kareem Darwish, Nizar Habash, Mourad Abbas, Hend Al-Khalifa, Huseein T. Al-Natsheh, Houda Bouamor, Karim Bouzoubaa, Violetta Cavalli-Sforza, Samhaa R. El-Beltagy, Wassim El-Hajj, Mustafa Jarrar, and Hamdy Mubarak. 2021. [A panoramic survey of natural language processing in the arab world](#). *Commun. ACM*, 64(4):72–81.
- Kareem Darwish and Walid Magdy. 2014. Arabic information retrieval. *Foundations and Trends in Information Retrieval*, 7(4):239–342.
- Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the fourteenth conference on computational natural language learning*, pages 107–116. Association for Computational Linguistics.
- María del Pilar Salas-Zárate, Mario Andrés Paredes-Valverde, Miguel Ángel Rodríguez-García, Rafael Valencia-García, and Giner Alor-Hernández. 2017. [Automatic detection of satire in twitter: A psycholinguistic-based approach](#). *Knowledge-Based Systems*, 128:20 – 33.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Fouad Soufiane Douzidia and Guy Lapalme. 2004. Lakhas, an arabic summarization system. In *Proceedings of DUC*, volume 4, pages 128–135.
- Megan L. Dress, Roger J. Kreuz, Kristen E. Link, and Gina M. Caucci. 2008. [Regional variation in the use of sarcasm](#). *Journal of Language and Social Psychology*, 27(1):71–85.

- Samhaa R El-Beltagy and Ahmed Ali. 2013. Open issues in the sentiment analysis of arabic social media: A case study. In *Innovations in information technology (iit), 2013 9th international conference on*, pages 215–220. IEEE.
- Samhaa R. El-Beltagy, Mona El Kalamawy, and Abu Bakr Soliman. 2017. [NileTMRG at SemEval-2017 task 4: Arabic sentiment analysis](#). In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 790–795, Vancouver, Canada. Association for Computational Linguistics.
- Abdelkader El Mahdaouy, Abdellah El Mekki, Kabil Essefar, Nabil El Mamoun, Ismail Berrada, and Ahmed Khoumsi. 2021. [Deep multi-task model for sarcasm detection and sentiment analysis in Arabic language](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 334–339, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Abdelkader El Mahdaouy, Abdellah El Mekki, Kabil Essefar, Abderrahman Skiredj, and Ismail Berrada. 2022. [CS-UM6P at SemEval-2022 task 6: Transformer-based models for intended sarcasm detection in English and Arabic](#). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 844–850, Seattle, United States. Association for Computational Linguistics.
- Heba Elfardy, Mohamed Al-Badrashiny, and Mona Diab. 2014. [AIDA: Identifying code switching in informal Arabic text](#). In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 94–101, Doha, Qatar. Association for Computational Linguistics.
- Hazem Elgabry, Shimaa Attia, Ahmed Abdel-Rahman, Ahmed Abdel-Ate, and Sandra Girgis. 2021. [A contextual word embedding for Arabic sarcasm detection with random forests](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 340–344, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Dalya Faraj, Dalya Faraj, and Malak Abdullah. 2021. [SarcasmDet at sarcasm detection task 2021 in Arabic using AraBERT pretrained model](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 345–350, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Elena Filatova. 2012. [Irony and sarcasm: Corpus generation and analysis using crowdsourcing](#). In *Proceedings of the Eighth International Conference on Language Re-*

- sources and Evaluation (LREC'12)*, pages 392–398, Istanbul, Turkey. European Language Resources Association (ELRA).
- Kamel Gaanoun and Imade Benelallam. 2021. [Sarcasm and sentiment detection in Arabic language a hybrid approach combining embeddings and rule-based features](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 351–356, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Bilal Ghanem, Jihen Karoui, Farah Benamara, Véronique Moriceau, and Paolo Rosso. 2019. Idat at fire2019: Overview of the track on irony detection in arabic tweets. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, pages 10–13.
- Bilal Ghanem, Jihen Karoui, Farah Benamara, Paolo Rosso, and Véronique Moriceau. 2020. Irony detection in a multilingual context. In *Advances in Information Retrieval*, pages 141–149, Cham. Springer International Publishing.
- Aniruddha Ghosh and Tony Veale. 2016. [Fracking sarcasm using neural network](#). In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 161–169, San Diego, California. Association for Computational Linguistics.
- Debanjan Ghosh, Weiwei Guo, and Smaranda Muresan. 2015. [Sarcastic or not: Word embeddings to predict the literal or sarcastic meaning of words](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1003–1012, Lisbon, Portugal. Association for Computational Linguistics.
- Debanjan Ghosh, Avijit Vajpayee, and Smaranda Muresan. 2020. [A report on the 2020 sarcasm detection shared task](#). In *Proceedings of the Second Workshop on Figurative Language Processing*, pages 1–11, Online. Association for Computational Linguistics.
- Dhaou Ghouil and Gaël Lejeune. 2021. [Sarcasm and sentiment detection in Arabic: investigating the interest of character-level features](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 329–333, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Raymond W. Gibbs. 1994. *The poetics of mind : figurative thought, language, and understanding*. Cambridge University Press, Cambridge.

- Raymond W. Gibbs. 2000. [Irony in talk among friends](#). *Metaphor and Symbol*, 15(1-2):5–27.
- H Paul Grice. 1975. Logic and conversation. In Peter Cole and Jerry L. Morgan, editors, *Syntax and Semantics: Vol. 3: Speech Acts*, pages 41–58. Academic Press.
- H Paul Grice, Peter Cole, and Jerry L Morgan. 1975. *Syntax and semantics*.
- Nizar Habash and Owen Rambow. 2005. [Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop](#). In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 573–580, Ann Arbor, Michigan. Association for Computational Linguistics.
- Nizar Habash, Owen Rambow, Mona Diab, and Reem Kanjawi-Faraj. 2008. Guidelines for annotation of arabic dialectness. In *Proceedings of the LREC Workshop on HLT & NLP within the Arabic world*, pages 49–53.
- Nizar Habash and Fatiha Sadat. 2006. [Arabic preprocessing schemes for statistical machine translation](#). In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 49–52, New York City, USA. Association for Computational Linguistics.
- Nizar Y Habash. 2010. Introduction to Arabic Natural Language Processing. *Synthesis Lectures on Human Language Technologies*, 3(1):1–187.
- Ali Hamdi, Khaled Shaban, and Anazida Zainal. 2016. A review on challenging issues in arabic sentiment analysis. *Journal of Computer Science*, 12(9):471–481.
- Devamanyu Hazarika, Soujanya Poria, Sruthi Gorantla, Erik Cambria, Roger Zimmermann, and Rada Mihalcea. 2018. [CASCADE: Contextual sarcasm detection in online discussion forums](#). In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1837–1848, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Amey Hengle, Atharva Kshirsagar, Shaily Desai, and Manisha Marathe. 2021. [Combining context-free and contextualized representations for Arabic sarcasm detection and sentiment identification](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 357–363, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

- Jeremy Howard and Sebastian Ruder. 2018. [Universal language model fine-tuning for text classification](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.
- Fatemah Husain and Ozlem Uzuner. 2021. [Leveraging offensive language for sarcasm and sentiment detection in Arabic](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 364–369, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Doaa Mohey El-Din Mohamed Hussein. 2018. [A survey on sentiment analysis challenges](#). *Journal of King Saud University - Engineering Sciences*, 30(4):330 – 338.
- Go Inoue, Bashar Alhafni, Nurpeiis Baimukan, Houda Bouamor, and Nizar Habash. 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*, pages 92–104, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Abraham Israeli, Yotam Nahum, Shai Fine, and Kfir Bar. 2021. [The IDC system for sentiment classification and sarcasm detection in Arabic](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 370–375, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Mohammed Jabreel and Antonio Moreno. 2017. [SiTAKA at SemEval-2017 task 4: Sentiment analysis in Twitter based on a rich set of features](#). In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 694–699, Vancouver, Canada. Association for Computational Linguistics.
- Julia Jorgensen. 1996. [The functions of sarcastic irony in speech](#). *Journal of Pragmatics*, 26(5):613–634.
- Aditya Joshi, Samarth Agrawal, Pushpak Bhattacharyya, and Mark J. Carman. 2018. Expect the unexpected: Harnessing sentence completion for sarcasm detection. In *Computational Linguistics*, pages 275–287, Singapore. Springer Singapore.
- Aditya Joshi, Pushpak Bhattacharyya, and Mark J Carman. 2017. Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50(5):73.
- Aditya Joshi, Vinita Sharma, and Pushpak Bhattacharyya. 2015. [Harnessing context incongruity for sarcasm detection](#). In *Proceedings of the 53rd Annual Meeting of*

the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 757–762, Beijing, China. Association for Computational Linguistics.

Aditya Joshi, Vaibhav Tripathi, Pushpak Bhattacharyya, and Mark J. Carman. 2016a. [Harnessing sequence labeling for sarcasm detection in dialogue from TV series ‘Friends’](#). In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 146–155, Berlin, Germany. Association for Computational Linguistics.

Aditya Joshi, Vaibhav Tripathi, Kevin Patel, Pushpak Bhattacharyya, and Mark Carman. 2016b. [Are word embedding-based features useful for sarcasm detection?](#) In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1006–1011, Austin, Texas. Association for Computational Linguistics.

Raquel Justo, Thomas Corcoran, Stephanie M Lukin, Marilyn Walker, and M Inés Torres. 2014. Extracting relevant knowledge for the detection of sarcasm and nastiness in the social web. *Knowledge-Based Systems*, 69:124–133.

Jihen Karoui, Farah Banamara Zitoune, and Veronique Moriceau. 2017. Soukhria: Towards an irony detection system for arabic in social media. *Procedia Computer Science*, 117:161–168.

Muhammad Khalifa and Noura Hussein. 2019. Ensemble learning for irony detection in arabic tweets. In *FIRE*.

Anupam Khattri, Aditya Joshi, Pushpak Bhattacharyya, and Mark Carman. 2015. [Your sentiment precedes you: Using an author’s historical tweets to predict sarcasm](#). In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 25–30, Lisboa, Portugal. Association for Computational Linguistics.

Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. 2018. [A large self-annotated corpus for sarcasm](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, pages 1–15.
- Svetlana Kiritchenko, Saif Mohammad, and Mohammad Salameh. 2016. [SemEval-2016 task 7: Determining sentiment intensity of English and Arabic phrases](#). In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 42–51, San Diego, California. Association for Computational Linguistics.
- Taku Kudo. 2018. [Subword regularization: Improving neural network translation models with multiple subword candidates](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.
- KUIS-AI-Lab. [Arabic-albert](#).
- Wuwei Lan, Yang Chen, Wei Xu, and Alan Ritter. 2020a. [An empirical study of pre-trained transformers for Arabic information extraction](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4727–4734, Online. Association for Computational Linguistics.
- Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020b. [Albert: A lite bert for self-supervised learning of language representations](#). In *International Conference on Learning Representations*.
- Joanne Larson, Timothy Morris, and Kristen Shaw. 2019. [Sarcasm as pedagogy of love: Exploring ironic speech acts in an urban high school english classroom](#). *Journal of Adolescent & Adult Literacy*, 63(2):169–177.
- Mohamed Lichouri, Mourad Abbas, Besma Benaziz, Aicha Zitouni, and Khaled Lounnas. 2021. [Preprocessing solutions for detection of sarcasm and sentiment for Arabic](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 376–380, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.
- Peng Liu, Wei Chen, Gaoyan Ou, Tengjiao Wang, Dongqing Yang, and Kai Lei. 2014. Sarcasm detection in social media based on imbalanced classification. In *Web-Age Information Management*, Cham. Springer International Publishing.

- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [Roberta: A robustly optimized BERT pretraining approach](#). *CoRR*, abs/1907.11692.
- Aya Lotfy, Marwan Torki, and Nagwa El-Makky. 2022. [AlexU-AL at SemEval-2022 task 6: Detecting sarcasm in Arabic text using deep learning techniques](#). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 891–895, Seattle, United States. Association for Computational Linguistics.
- Stephanie Lukin and Marilyn Walker. 2013. [Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue](#). In *Proceedings of the Workshop on Language Analysis in Social Media*, pages 30–40, Atlanta, Georgia. Association for Computational Linguistics.
- Quinn McNemar. 1947. [Note on the sampling error of the difference between correlated proportions or percentages](#). *Psychometrika*, 12(2):153–157.
- Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. [context2vec: Learning generic context embedding with bidirectional LSTM](#). In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 51–61, Berlin, Germany. Association for Computational Linguistics.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations*, pages 1–12.
- Ahmed Mourad and Kareem Darwish. 2013. [Subjectivity and sentiment analysis of Modern Standard Arabic and Arabic microblogs](#). In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 55–64, Atlanta, Georgia. Association for Computational Linguistics.
- Hamdy Mubarak, Kareem Darwish, Walid Magdy, Tamer Elsayed, and Hend Al-Khalifa. 2020. [Overview of OSACT4 Arabic offensive language detection shared task](#). In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 48–52, Marseille, France. European Language Resource Association.

- Mahmoud Nabil, Mohamed Aly, and Amir Atiya. 2015. [ASTD: Arabic sentiment tweets dataset](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2515–2519, Lisbon, Portugal. Association for Computational Linguistics.
- Maryam Najafi and Ehsan Tavan. 2022. [MarSan at SemEval-2022 task 6: iSarcasm detection via t5 and sequence learners](#). In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 978–986, Seattle, United States. Association for Computational Linguistics.
- Malek Naski, Abir Messaoudi, Hatem Haddad, Moez BenHajhmida, Chayma Fourati, and Aymen Ben Elhaj Mabrouk. 2021. [iCompass at shared task on sarcasm and sentiment detection in Arabic](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 381–385, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Hamada Nayel, Eslam Amer, Aya Allam, and Hanya Abdallah. 2021. [Machine learning-based model for sentiment and sarcasm detection](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 386–389, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Silviu Oprea and Walid Magdy. 2019. [Exploring author context for detecting intended vs perceived sarcasm](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2859, Florence, Italy. Association for Computational Linguistics.
- Silviu Oprea and Walid Magdy. 2020a. [iSarcasm: A dataset of intended sarcasm](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1279–1289, Online. Association for Computational Linguistics.
- Silviu Vlad Oprea and Walid Magdy. 2020b. [The effect of sociocultural variables on sarcasm communication online](#). *Proc. ACM Hum.-Comput. Interact.*, 4(CSCW1).
- Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2020. [A monolingual approach to contextualized word embeddings for mid-resource languages](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online. Association for Computational Linguistics.

- Ahmed Oussous, Fatima-Zahra Benjelloun, Ayoub Ait Lahcen, and Samir Belfkih. 2020. Asa: A framework for arabic sentiment analysis. *Journal of Information Science*, 46(4):544–559.
- Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. [Thumbs up? sentiment classification using machine learning techniques](#). In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. [GloVe: Global vectors for word representation](#). In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 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)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- Louise H Phillips, Roy Allen, Rebecca Bull, Alexandra Hering, Matthias Kliegel, and Shelley Channon. 2015. Older adults have difficulty in decoding sarcasm. *Developmental psychology*, 51(12):1840.
- Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. [SemEval-2016 task 5: Aspect based sentiment analysis](#). In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California. Association for Computational Linguistics.
- Tomáš Ptáček, Ivan Habernal, and Jun Hong. 2014. [Sarcasm detection on Czech and English Twitter](#). In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 213–223, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

- Veranika Puhacheuskaya and Juhani Järvikivi. 2022. I was being sarcastic!: The effect of foreign accent and political ideology on irony (mis) understanding. *Acta Psychologica*, 222:103479.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020a. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020b. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Ashwin Rajadesingan, Reza Zafarani, and Huan Liu. 2015. Sarcasm detection on twitter: A behavioral modeling approach. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 97–106. ACM.
- Eshrag Refaee and Verena Rieser. 2014. An arabic twitter corpus for subjectivity and sentiment analysis. In *LREC*, pages 2268–2273.
- Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. 2013. [Sarcasm as contrast between a positive sentiment and negative situation](#). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 704–714, Seattle, Washington, USA. Association for Computational Linguistics.
- Patricia Rockwell and Evelyn M. Theriot. 2001. [Culture, gender, and gender mix in encoders of sarcasm: A self-assessment analysis](#). *Communication Research Reports*, 18(1):44–52.
- Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. [SemEval-2017 task 4: Sentiment analysis in Twitter](#). In *Proceedings of the 11th International Workshop on*

- Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada. Association for Computational Linguistics.
- Paul Rottger, Bertie Vidgen, Dirk Hovy, and Janet Pierrehumbert. 2022. [Two contrasting data annotation paradigms for subjective NLP tasks](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 175–190, Seattle, United States. Association for Computational Linguistics.
- Ali Safaya, Moutasem Abdullatif, and Deniz Yuret. 2020. [KUISAIL at SemEval-2020 task 12: BERT-CNN for offensive speech identification in social media](#). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2054–2059, Barcelona (online). International Committee for Computational Linguistics.
- Boaz Shmueli, Lun-Wei Ku, and Soumya Ray. 2020. [Reactive Supervision: A New Method for Collecting Sarcasm Data](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2553–2559, Online. Association for Computational Linguistics.
- Taysir Hassan Soliman, M. A. Elmasry, A. Hedar, and M. M. Doss. 2014. Sentiment analysis of arabic slang comments on facebook. *International Journal of Computers & Technology*, 12(5):3470–3478.
- Bingyan Song, Chunguang Pan, Shengguang Wang, and Zhipeng Luo. 2021. [Deep-BlueAI at WANLP-EACL2021 task 2: A deep ensemble-based method for sarcasm and sentiment detection in Arabic](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 390–394, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Dan Sperber and Deirdre Wilson. 1981. Irony and the use-mention distinction. *Philosophy*, 3:143–184.
- Yi Tay, Anh Tuan Luu, Siu Cheung Hui, and Jian Su. 2018. [Reasoning with sarcasm by reading in-between](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1010–1020, Melbourne, Australia. Association for Computational Linguistics.
- Akira Utsumi. 2000. Verbal irony as implicit display of ironic environment: Distinguishing ironic utterances from nonirony. *Journal of Pragmatics*, 32(12):1777–1806.

- Tim Van de Cruys and Begoña Villada Moirón. 2007. [Semantics-based multiword expression extraction](#). In *Proceedings of the Workshop on A Broader Perspective on Multiword Expressions*, pages 25–32, Prague, Czech Republic. Association for Computational Linguistics.
- Cynthia Van Hee, Els Lefever, and Véronique Hoste. 2018. [SemEval-2018 task 3: Irony detection in English tweets](#). In *Proceedings of the 12th International Workshop on Semantic Evaluation*, pages 39–50, New Orleans, Louisiana. Association for Computational Linguistics.
- Anshul Wadhawan. 2021. [AraBERT and farasa segmentation based approach for sarcasm and sentiment detection in Arabic tweets](#). In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 395–400, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.
- Marilyn Walker, Jean Fox Tree, Pranav Anand, Rob Abbott, and Joseph King. 2012. [A corpus for research on deliberation and debate](#). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 812–817, Istanbul, Turkey. European Language Resources Association (ELRA).
- Byron C. Wallace, Do Kook Choe, and Eugene Charniak. 2015. [Sparse, contextually informed models for irony detection: Exploiting user communities, entities and sentiment](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1035–1044, Beijing, China. Association for Computational Linguistics.
- Byron C. Wallace, Do Kook Choe, Laura Kertz, and Eugene Charniak. 2014. [Humans require context to infer ironic intent \(so computers probably do, too\)](#). In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 512–516, Baltimore, Maryland. Association for Computational Linguistics.
- Deirdre Wilson. 2006. The pragmatics of verbal irony: Echo or pretence? *Lingua*, 116(10):1722–1743.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R’emi Louf, Morgan Funtowicz, and Jamie

- Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.
- Omar F. Zaidan and Chris Callison-Burch. 2014. [Arabic Dialect Identification](#). *Computational Linguistics*, 40(1):171–202.
- Andrea Zaninello and Alexandra Birch. 2020. [Multiword expression aware neural machine translation](#). In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3816–3825, Marseille, France. European Language Resources Association.
- Ayman A Zayyan, Mohamed Elmahdy, Husniza binti Husni, and Jihad M Al Ja'am. 2016. Automatic diacritics restoration for dialectal arabic text. *International Journal of Computing & Information Sciences*, 12(2).
- Chiyu Zhang and Muhammad Abdul-Mageed. 2019. Multi-task bidirectional transformer representations for irony detection. *ArXiv*, abs/1909.03526.
- Meishan Zhang, Yue Zhang, and Guohong Fu. 2016. [Tweet sarcasm detection using deep neural network](#). In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2449–2460, Osaka, Japan. The COLING 2016 Organizing Committee.

Arabic Bibliography

1. عندليب ، علي و فرع شيرازي ، السيد حيدر. 2019. صراع المفارقة بين السخرية والمناقضة دراسة لغوية ودلالية. بحوث في اللغة العربية، 11(20)، 121-13.
2. داودي، صورية وشارف، عماد. 2019. آليات اشتغال المفارقة وجمالياتها الشعرية في ديوان لافتات لأحمد مطر. مجلة إشكالات في اللغة و الأدب، 8، 520-500
3. المنصوري، حافظ كوزي عبد العالي. 2012. السخرية الهادفة في شعر أحمد مطر. مجلة اللغة العربية و آدابها، 2012(15)، 149-194
4. إبراهيم، نبيلة. 1987. المفارقة. فصول، 7(4) ، 137
5. العبودي، ضياء غني لفنة و عبد الله حسين خلف. 2014. الأمانى و البعد المفارقة في نقائض جرير و الفرزدق. مجلة جامعة ابن رشد في هولندا، 2014(12) ، 2-22
6. رجب، صالح حسن. 2009. المفارقة في شعر محمود درويش ديوان "عاشق من فلسطين" نموذجا. مجلة جامعة الأزهر، سلسلة العلوم الإنسانية، 11

English Translation

1. Andalib, Ali and Alsayed Haidar Fare' Shirazi. 2019. Controversy over the Concept of Irony (=Al-Mophareqeh) from Sarcasm to Contradiction; A Linguistic and Semantic Approach. Research in Arabic Language, 11(20), 121-134.
2. Daoudi, Sourieh and Sharif, Imad. 2019. Mechanisms of Irony and its Poetic Beauty in Ahmed Matar's Lafit. Journal of Issues in Language and Literature, 8, 500-520
3. Al-Mansoori, Hafez Kozi Abdel-Aali. 2012. Purposeful Irony in Ahmad Matar's Poetry. Journal of Arabic Language and Literature, 2012(15), 149-194

4. Ibrahim, Nabila. 1987. The Irony. *Fusool*, 7(4), 137
5. Al-Aboudi, Daa Ghani Lafta and Abdullah Hussein Khalaf. 2014. Aspirations and the Ironic Dimension in the Antithesis of Jarir and Al-Farazdaq. *Journal of Ibn Rushd University in the Netherlands*, 2014(12), 2-22
6. Ragab, Salih Hassan. 2009. The Irony in Mahmoud Darwish's Poetry "A Lover from Palestine" as a Model. *Al-Azhar University Journal, Human Sciences Series*, 11

Appendix A

Annotation Guidelines

هدف البحث: تجميع المعلومات لتطوير التحليل الاوتوماتيكي للعواطف و الآراء.

الفوائد: هذا البحث لن يفيدك بشكل مباشر لكن سوف يفيد المجتمع بشكل عام من حيث تطوير كيفية تحليل الكمبيوتر للغات البشرية. قد يؤدي ذلك الى تطوير برامج تحليل اراء وأحاسيس متطورة، بالإضافة الى تطوير بحث الانترنت. هذه المهمة تهدف إلى تصنيف التغريدات (تويت) بناء على ما تحتويه من عواطف (مشاعر) إلى تغريدات ذات محتوى إيجابي أو سلبي أو محايد، بالإضافة إلى تصنيفها في حال احتوت على سخرية أو تهكم. بالإضافة إلى ذلك سوف يقوم المشاركون باختيار الى اي لهجة تنتمي هذه التغريدة.

الخطوات:

1. قم بقراءة التغريدة
2. قم باختيار نوع الشعور في التغريدة (إيجابي ، سلبي ، محايد) حيث نص السؤال (sentiment)
3. قم باختيار فيما إذا كانت التغريدة تحتوي على تهكم أو سخرية حيث نص السؤال (sarcasm)
4. اختيار لهجة التغريدة حيث نص السؤال (dialect)

أمثلة:

النصوص الإيجابية هي التي تحتوي بطابعها شعورا إيجابيا تهنئة أو مناسبة أو ربح أو أي حدث يبعث أو يبشر بالخير و التفاؤل.

1. تصفيات كأس العالم سويسرا و مدري مين المهم مباريات حلوووه
2. محمد صلاح يستحق افضل لاعب

النص السلبي هو الذي يعبر عن شيء سلبي أو محزن أو أي خبر سيء أو سلبي بالإضافة الى التعبير عن الغضب او الانفعال. فيما يلي بعض الأمثلة:

1. مفيش حماس خالص في تصفيات كأس العالم مش حاسس باي حاجة خالص
2. انا مع مقاطعة الانتخابات ما لم يكن هناك مسار ثوري حقيقي يحقق اهداف ثورة يناير

النص الحيادي (المحايد) هو الذي لا يحتوي على تعبيرات إيجابية أو سلبية. فيما يلي بعض الأمثلة:

1. هو لعب كام ماتش في تصفيات كأس العالم؟
2. الرئيس السيسي: الغرب يعتقد اننا ضد حقوق الانسان . و اقول لهم : لا.. نحن امة تريد العيش بسلام

فيما يتعلق السخرية أو التهكم فتكون عندما يكون المقصود بالنص عكس المكتوب ويكون الغرض هو السخرية من شيء بطريقة غير مباشر. فيما يلي بعض الأمثلة:

1. التصفيات اللي قعد اسويها بحياتي اقوي من تصفيات كأس العالم
2. ابني شايل 3 مواد في الجامعة. حاجة تشرف

Figure 1.1: A sample of the guidelines provided to the annotators.

Appendix B

Arabic iSarcasmEval Data Collection Instructions

فهم أساليب التعبير في اللغة العربية
تهدف هذه الدراسة إلى فهم أساليب التعبير غير المباشر في اللغة العربية. هذا الأمر يشتمل على عدة أمور منها السخرية أو التشبيه.

في هذه التجربة نحتاج من كل شخص أن يقوم بإدخال مثال على جملة أو منشور ساخر بالإضافة إلى توضيح السخرية المقصودة في هذه الجملة. لا يشترط أن يكون النص باللغة القصصى ويفضل أن يكون بالعامة.

يمكن للمشاركة نسخ منشور من أي مكان ولا يشترط تأليفه.

بعد الانتهاء من الاستبيان سيحصل المشترك على رمز (كود)، يرجى لصقه في المربع النصي في الأسفل.

يجب على كل مشترك إدخال 5 جمل مختلفة مع كامل تفاصيلها كما هو موضح في الأمثلة التالية:

الموضوع	الشعور	اللهجة	التوضيح	الجزء الساخر	النص بطريقة مباشرة	النص
غير ذلك	سلبى	مصري	ده غبي لكن فاكر نفسه عيقرى زي العالم اينشتاين	اينشتاين	ده غبي	ده فاكر نفسه اينشتاين 😏
غير ذلك	سلبى	شامي	مسخرة انه الشخص اللي ما بعمالك قيمة هو غير مخلص	مخلص	اللي بحسبك انه بغيالك رح بجمي غيرك هاد انسان خاين	مخلص جدا اللي بحسبك انه بغيالك رح بجمي حد مكاك 😏
رياضة	سلبى	خليجي (سعودية)	حتى يتحاشى الإحراج و ينسحب من الموقف بطريقة لا تبين ردة فعل من الممكن تحليلها.	سوي نفسك ميت	مدرب الفريق المنافس حزين و لا يريد ان يظهر ذلك للعلن.	الف ميروك الانتصار لفريقنا ، مدرب الفريق الثاني ماخذ وضعية سوي نفسك ميت !
رياضة	محايد	فصحى	سخرية من محمد صلاح و الصاق امور خيالية كإنفاذ سمكة لحسن اخلاقه	ينقذ سمكة من الفرق	محمد صلاح يحمل سمكة	محمد صلاح ينقذ سمكة من الفرق، الله على اخلاقك يا فخر العرب
سياسي	سلبى	شامي	مسخرة من سوء ترامب و انه فاش زيه	فش زيه	هاظ ترامب فاش اسوأ منه	هاظ ترامب فاش زيه، تحفة 🤡

Figure 2.1: A snippet of the instructions used to collect the Arabic iSarcasmEval dataset.

Appendix C

Results on Sentiment Analysis Task

Model	Recall	Accuracy	F _{PN}
BiLSTM	0.623	0.671	0.691
mBERT	0.570	0.624	0.625
GigaBERT	0.625	0.662	0.673
XLM-R-base	0.605	0.643	0.661
XLM-R-large	0.641	0.678	0.699
<hr style="border-top: 1px dashed black;"/>			
AraBERT-base (v01)	0.630	0.670	0.691
AraBERT-base (v1)	0.638	0.677	0.696
AraBERT-base (v02)	0.654	0.686	0.709
AraBERT-base (v2)	0.651	0.690	0.711
AraBERT-large (v02)	0.659	0.695	0.718
AraBERT-large (v2)	0.660	0.700	0.724
AraELECTRA (discriminator)	0.649	0.687	0.709
AraELECTRA (generator)	0.604	0.648	0.663
Arabic BERT-base	0.627	0.668	0.687
Arabic BERT-large	0.648	0.678	0.699
Arabic ALBERT-base	0.600	0.653	0.663
Arabic ALBERT-large	0.603	0.657	0.669
Arabic ALBERT-xlarge	0.623	0.674	0.691
MARBERT	0.664	0.693	0.724
ARBERT	0.642	0.673	0.695
QARiB	0.661	0.688	0.720
AraGPT2-base	0.594	0.647	0.662
AraGPT2-medium	0.602	0.649	0.666
AraGPT2-large	0.562	0.612	0.629

Table 3.1: Results achieved by all models described in Chapter 5 on the sentiment analysis task in ArSarcasm-v2 dataset.

Appendix D

ChatGPT Experiments

To analyse the performance of advanced models on the sarcasm detection task, we tested ChatGPT on some of the examples provided in Section 6.1.3. We asked the model to identify whether a text is sarcastic and justify the answer. We started with the following prompt:

You are a smart assistant who understands Arabic. You are an expert in Arabic sarcasm. When provided an Arabic sentence, please answer whether it is sarcastic and explain why. Your answer should be in Arabic and English using the following format:

O: original sentence

S: sarcastic or not

E: explanation

O_en: English translation of the original sentence

S_en: sarcastic or not (in English)

E_en: Explanation in English

Figure 4.1 shows ChatGPT's answer when prompted with a sarcastic example. The answer is incorrect as the sentence is sarcastic and used to express being annoyed by the community and the irrational behaviour of some people. Figure 4.2 shows a correct answer from ChatGPT. However, this is limited to the answer of whether the text is sarcastic. The translation and the explanation are incorrect. The accurate translation is "those who don't know, say lentil". The Arabic sentence is a reference to a proverb to express disappointment in people who are fast to judge without knowing the details.

O: شعب يتعاطى كلور
 S: غير ساخرة
 E: تعبر الجملة عن ادعاء بأن الشعب يتعاطى مادة الكلور بصفة دائمة وكميات كبيرة. وهو تصريح مبالغ فيه لتوضيح خطورة تعاطي المواد الكيميائية و السامة.
 O_en: People always consume chlorine
 S_en: Not sarcastic
 E_en: The sentence is claiming that people constantly consume chlorine in large amounts, which is an exaggeration intended to emphasize the danger of consuming chemicals and toxins.

Figure 4.1: Incorrect ChatGPT answer for a sarcastic example.

O: الي ميعرفش يقول عدس
 S: ساخرة
 E: الجملة ساخرة وتستخدم عادةً للتعبير عن أن الشخص الذي لا يعرف شيئاً عن موضوع ما يتظاهر بأنه خبير بهذا الموضوع ويتحدث عنه بطريقة عشوائية وغير منطقية. يستخدم الأسلوب الساخر للتعبير عن السخرية والاستهزاء من هذا الشخص.
 O_en: Someone who doesn't know claims to be an expert on lentils
 S_en: Sarcastic
 E_en: The sentence is sarcastic and is commonly used to express that a person who knows nothing about a certain topic pretends to be an expert and talks about it in a random and illogical way. The sarcastic tone is used to express ridicule and mockery towards this person.

Figure 4.2: Correct ChatGPT answer for a sarcastic example. Incorrect explanation.

Figure 4.3 shows a correct answer from ChatGPT regarding whether the sentence is sarcastic. However, the provided translation and explanation are incorrect. The accurate translation is “He is the size of a mouse”. This sentence is used to ridicule someone’s size or height.

Figure 4.4 shows an incorrect answer from the model. The model incorrectly assumed the sentence to be sarcastic. The sentence expresses disappointment from a person who is overthinking and wishes to think less. In the explanation, the model suggests another implied understanding of the sentences, which is the same as the original one.

O: قده قد الفأرة :
 S: ساخرة
 E: الجملة ساخرة وتستخدم لوصف شخص يتظاهر بالقوة والشجاعة، بالرغم من أنه ضعيف وجبان. تستخدم الأسلوب الساخر للتعبير عن السخرية والاستهزاء من هذا الشخص الذي يتظاهر بالقوة والشجاعة ولكن في الحقيقة ليس كذلك.
 O_en: He's as brave as a mouse
 S_en: Sarcastic
 E_en: The sentence is sarcastic and is used to describe someone who pretends to be strong and brave, despite being weak and cowardly. The sarcastic tone is used to express ridicule and mockery towards this person who pretends to be strong and brave, but in reality is not.

Figure 4.3: Correct ChatGPT answer for a sarcastic example. Incorrect explanation.

O: ليت اكون شخص ما يفكر كثير :
 S: ساخرة
 E: الجملة ساخرة وتستخدم للتعبير عن الإحباط وعدم الرضا بالنفس، ويمكن فهمها على أنها تقول بالعكس تمامًا، حيث أنها تعني "أتمنى أن لا أكون شخصًا يفكر كثيرًا" وبذلك يتم الإشارة إلى أن التفكير هو شيء جيد ومطلوب ولا ينبغي الاستسلام للعزلة والجهل. تستخدم الأسلوب الساخر للتعبير عن التناقض بين ما يتمناه الشخص وما هو في الواقع ضروري ومهم.
 O_en: I wish I was a person who doesn't think too much
 S_en: Sarcastic
 E_en: The sentence is sarcastic and is used to express frustration and dissatisfaction with oneself. It can be understood as "I wish I wasn't someone who thinks too much", thus implying that thinking is a good and necessary thing and one should not succumb to ignorance and isolation. The sarcastic tone is used to express the contradiction between what the person wishes for and what is actually important and necessary.

Figure 4.4: Inorrect ChatGPT answer for a non-sarcastic example.

Figure 4.5 shows that the model correctly identified that the sentence is non-sarcastic. However, the model's translation is inaccurate, and the more accurate one is "I love to be asked about all the details". The inaccurate translation could be the reason for the incorrect explanation. Figure 4.6, shows that the model incorrectly predicted the sentence to be sarcastic. The sentence is indeed used to express surprise at such behaviour, but it is not sarcastic.

O: أحب لما أتسأل عن كل التفاصيل
 S: غير ساخرة
 E: الجملة غير ساخرة وتعبر عن حب الشخص لمعرفة كل التفاصيل والمعلومات المتعلقة بموضوع ما، ويمكن أن يكون ذلك بسبب الفضول أو الحاجة للمعرفة أو الاطمئنان. تعبر هذه الجملة عن الرغبة في الحصول على المزيد من المعلومات وتفادي ترك أي تفاصيل غير معروفة أو غامضة.
 O_en: I love asking about all the details
 S_en: Not sarcastic
 E_en: The sentence is not sarcastic and expresses a person's love for knowing all the details and information related to a certain topic. This may be due to curiosity, the need for knowledge, or reassurance. This sentence expresses the desire to obtain more information and avoid leaving any unknown or mysterious details.

Figure 4.5: Correct ChatGPT answer for a non-sarcastic example. Incorrect explanation and translation.

O: جماعه ممكن لما حد يضايقي انا اللي ازعل مش هو؟
 S: ساخرة
 E: الجملة ساخرة وتعبر عن الاستغراب والتعجب من شخص يظهر انزعاجه وغضبه عندما يتعرض لإزعاج من شخص آخر، ولكن في الحقيقة يعكس ذلك عدم القدرة على التحكم في مشاعره والتعامل معها بشكل صحيح. الأسلوب الساخر يستخدم لتعبير عن السخرية من هذا التصرف الذي يبدو ضعيفاً وغير محترم للمشاعر الحقيقية للشخص الذي يسبب الإزعاج.
 O_en: Is it possible that when someone annoys me, it's actually me who gets upset, not them?
 S_en: Sarcastic
 E_en: The sentence is sarcastic and expresses surprise and incredulity at a person who shows annoyance and anger when they are bothered by someone else, but in reality, it reflects their inability to control their emotions and deal with them properly. The sarcastic tone is used to express ridicule towards this behavior, which appears weak and disrespectful to the true feelings of the person causing the annoyance.

Figure 4.6: Incorrect ChatGPT answer for a non-sarcastic example.