SarcasmDet at Sarcasm Detection Task 2021 in Arabic using AraBERT Pretrained Model

Dalya Faraj
Department of Computer Science
Jordan University of Science
and Technology
Irbid, Jordan
dfalnore18@cit.just.edu.jo

Malak Abdullah
Department of Computer Science
Jordan University of Science
and Technology
Irbid, Jordan
mabdullah@just.edu.jo

Abstract

This paper presents one of the top five winning solutions for the Shared Task on Sarcasm and Sentiment Detection in Arabic (sub-task1 Sarcasm Detection). The goal of the sub-task is to identify whether a tweet is sarcastic or not. Our solution has been developed using ensemble technique with AraBERT pre-trained model. This paper describes the architecture of the submitted solution in the shared task. It also provides in detail the experiments and the hyperparameters tuning that lead to this outperforming result. Besides, the paper discusses and analyzes the results by comparing all the models that we have trained or tested to build a robust model in a table design. Our model is ranked fifth out of 27 teams with an F1-score of 0.5989 of the sarcastic class. It is worth mentioning that our model achieved the highest accuracy score of 0.7830 in this competition.

Keywords: Arabic, Sentiment Analysis, Sarcasem, NLP, AraBERT.

1 Introduction

Sarcasm, bullying, and offense are critical topics that dramatically increase in social media, leading to countless problems, whether on the personal or community level. Irony and sarcasm use positive words commonly applied in public life to make fun, ridicule, or annoy someone, either directly or indirectly (González-Ibánez et al., 2011). Detecting sarcasm depends on the sentence context and understanding the current situation, which can be challenging to be discovered using traditional techniques. With the development of artificial intelligence (AI) methods, specifically natural language processing (NLP), predicting and reducing such words that denote sarcasm are investigated thoroughly using sentiment and emotion analysis methodologies (Abdullah et al., 2018; Ogudo and Nestor, 2019; Cheng and Tsai, 2019).

Sentiment analysis (SA) refers to the use of natural language processing, text analysis, and linguistics to identify, extract, quantify, and systematically determine whether a sentence or a document is positive or negative (Duwairi and El-Orfali, 2014; Al-Ayyoub et al., 2019). SA techniques have been applied to utilize the process of predicting whether the text contains sarcasm and offensive words within it (Feldman, 2013).

The lack of resources and annotated datasets for the Arabic language is considered one of the challenges in this field (Abdullah and Hadzikadic, 2017; Abdulla et al., 2013). For detecting sarcasm, a group of researchers (Abu Farha et al., 2021) provided new datasets for the shared task on Sarcasm and Sentiment Detection in Arabic (WANLP 2021). The shared task includes two subtasks: the Sarcasm Detection sub-task for detecting if the tweet is sarcastic or not. The Sentiment Analysis sub-task, which is focused on analyzing tweets then identifying their sentiment.

This paper presents the SarcasemDet team model at the WANLP 2021 Shared Task for sub-task 1 (Sarcasm Detection). Our solution system (SarcasmDet) is one of the top five teams among 27 participated teams. The proposed approach uses the pre-trained language model AraBERT (Antoun et al.). We also have experimented with several pre-trained language models using the simple transformers library. It is worth mentioning that using the hard-voting ensemble technique has increased the performance of the model, remarkably.

The rest of the paper goes as follows. Section 2 overviews and discusses the existed work

Proceedings of the Sixth Arabic Natural Language Processing Workshop, pages 345–350
Kyiv, Ukraine (Virtual), April 19, 2021.

related to this research. Section 3 illustrates the shared task and describes the dataset. Section 4 displays our solution architecture and preprocessing method to achieve the best accuracy score in this task. Section 5 presents the experiments with hyperparameters tuning analysis. Section 6 presents the results. Finally, in section 7, the Conclusion.

2 Related Work

There are numerous resources on sarcasm detection in English and European languages, but there is a lack of resources and datasets in the Arabic language. The focus on detecting sarcasm in the Arabic language emerged in recent years. In (Karoui et al., 2017), the researchers were among the first people to provide a dataset to detect sarcasm in Arabic. They collected Arabic tweets of different Arabic dialects, such as Egyptian, Syrian and Saudi dialects. They cleaned the dataset, 5,479 tweets, including 1733 irony, and added four features for each tweet: surface, sentiment, shifter, and internal context features. The experiments showed that the Random Forest classifier achieved high accuracy with 72.76% to detect sarcasm in the Arabic language tweets.

Another sarcasm dataset in the Arabic language is provided by (Farha and Magdy, 2020). This dataset contains sarcasm, sentiment, and dialect labels incorporating 10,547 rows of tweets where 16% are sarcastic tweets. The researchers applied the Bidirectional LSTM (biLSTM) deep learning approach to achieve an F1-score of 0.46, which indicates the hardness of detecting sarcasm in Arabic.

Various machine learning mechanisms have been used to detect sarcasm and offensive in Arabic tweets. In (Al-Ghadhban et al., 2017), the authors used the supervised Naïve Bayes Multinomial Text algorithm to detect the sarcasm in Arabic tweets. They collected 344 rows of tweets data manually, where 238 out of them are sarcastic tweets, and 106 are nonsarcastic tweets. To evaluate the NB model, they used recall, precision, f1-score metrics, and the resultant outputs are 0.659, 0.710, and 0.676, respectively.

In (Hassan et al., 2020), the researchers participated in the SemEval 2020 Task 12 Arabic offensive language dataset for two subtasks: offensive language detection and hate speech detection. They were the winning solution system for discovering offensive language in the Arabic language. The dataset they used contains 10,000 rows of tweets data manually labeled for offensiveness (OFF or NOT OFF). They applied several deep and machine learning algorithms, such as DNN, CNN, RNN, and SVM. They achieved the best results with an F1-score of 90.51% by ensembling SVM and DNN.

3 Task and Dataset Description

In the Shared Task on Sarcasm and Sentiment Detection in Arabic from WANLP 2021(Abu Farha et al., 2021) challenge, all tasks have different requirements. The shared task includes two subtasks: subtask 1, Sarcasm Detection, which aims to detect if the tweet is sarcastic or not. Subtask 2, Sentiment Analysis, focused on analyzing tweets then identifying their sentiments. Table1 shows a sample of the training dataset. It is worth mentioning that we have participated in subtask 1.

#	**Example**	Sarcastic
290	#عرب _ولكن عربي :قهوته برازيلية بدلته ايطالية ساعته سويسريةوعطره فرنسي سيارته المانية.الخ ويذهب لالقاء محاضرته بعنوان #قاطعوا _الغرب	True
9029	من احتفالات المولد النبوي الشريف	False

Table 1: sample for subtask1 Sarcasm Detection from the training dataset

In our solution system, we have used the ArSarcasm-v2 dataset provided by (Abu Farha et al., 2021). The data is separated into two parts: a training dataset containing 12,548 tweets and four features labeled as follows: tweet, sarcasm, sentiment, and dialect. The second is the test dataset containing 3,000 tweets and two features labeled as a tweet and dialect. Table1 shows a description of the 15,548 rows of data. One of the challenges in this task is the imbalanced dataset that contains 2,168 sarcastic tweets and 10,380 nonsarcastic tweets. Figure 1 shows the distribution of classes in the training set. Data preprocessing used

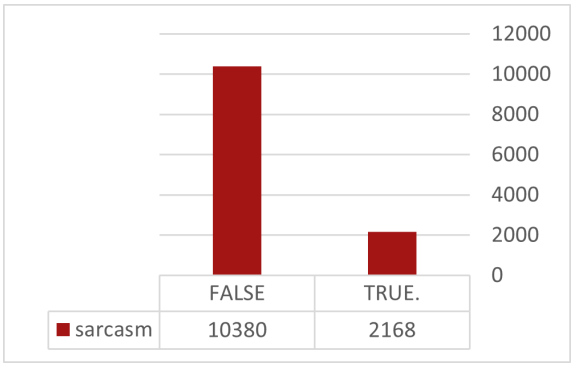

Figure 1: Classes Distribution in the training set for the sarcastic tweet labels.

are from the NLTK library (Bird, 2006), such as normalized duplicated characters and cleaning data from emojis, links, and Html tags.

Dataset	Size	sarcastic	non-sarcastic
Train	12548	2168	10380
Test	3000	821	2179
Total	**15548**	**2989**	**12559**

Table 2: The total size of the training and testing dataset, and an indication of the number if it's sarcastic tweets or not.

4 SarcasmDet Description

In our approach, we have used the Arabic pre-trained model called AraBERT (Antoun et al.). AraBert is a pre-trained model that focuses directly on the Arabic language, and it is based on the BERT architecture (Devlin et al., 2018). The AraBERT model's strength is being one of the best language models for NLP and sentiment analysis for the Arabic language. There are two versions of AraBERT(v01 and v02). The first version, AraBERT-v01, was trained on 77M sentences, with a size of 23GB and 2.7B of words. The second version, AraBERT-v02, was trained on 200M sentences with a size of 77GB and 8.6B words. In our approach, we have used both versions that are located in HuggingFace library(Wolf et al., 2019). We have experimented with several hyperparameters and the best result achieved from AraBERT-v02 is 0.5650 f1-score for the sarcastic class (f1-sarcastic), with learning_rate= 1e-5, manual_seed= 17, train_batch_size= 16 and num_train_epochs= 5. To enhance our results, we have used the ensemble technique, hard-voting (Chou et al., 2009). The best-ensembled model, SarcasmDet, achieved an f1-sarcastic (f1-score for sarcastic class) with 0.5989 in the testing phase. SarcasmDet has ensembled the AraBERT-v01 and AraBERT-v02 models. Figure 2 shows the architecture of SarcasmDet in detail.

5 Experiments

We have experimented with several deep learning models to detect sarcastic Arabic tweets through our training and testing phases. One of the experimented models is XLM-RoBERTa(XLM-R) (Liu et al., 2019), which performed less than other models in this task. The multilingual BERT (mBERT) model obtained one of the best results in our experiments because of multilingual support. The AraBERT was the one that captured the highest score with the v02 model. Table 3 shows the hyperparameters we have used in our experiments for the tested models.

All of the models have been implemented using the HuggingFace library and SimpleTransformer pre-trained package. Some models, such as XLM-R, have default learning_rate=6e, but we changed it to 1e-5, which increased the results.

6 Results and Discussion

Our solution was separated into two phases. The first phase was the development phase. We have experienced several deep learning models XLM-R, AraBERT(v01,v02), mBERT(cased/uncased) by implementing them into the dataset to make the solution that suits the next phase, which is the test phase. In the test phase, our model performed great by using the ensemble technique and was ranking in the top 5 models. We believe that The AraBERT-v02 model has an excellent understanding of the Arabic language and sentence analysis. We have noticed that, with or without data preprocessing and hard-voting ensemble techniques, the model outperformed other models remarkably. We expect the exceeding of the AraBERT-v02 since it is already trained and tested on a massive amount of Arabic data. In the test phase section, we showed our experiments and the results that we achieved. Figure3 shows a plot chart for the best results.

6.1 Test Phase

We run multiple experiments on the ArSarcasm-v2 dataset in the test phase using AraBERT, XLM-R, and mBERT models. We chose AraBERT-v02 because this model scored the best F1-score with

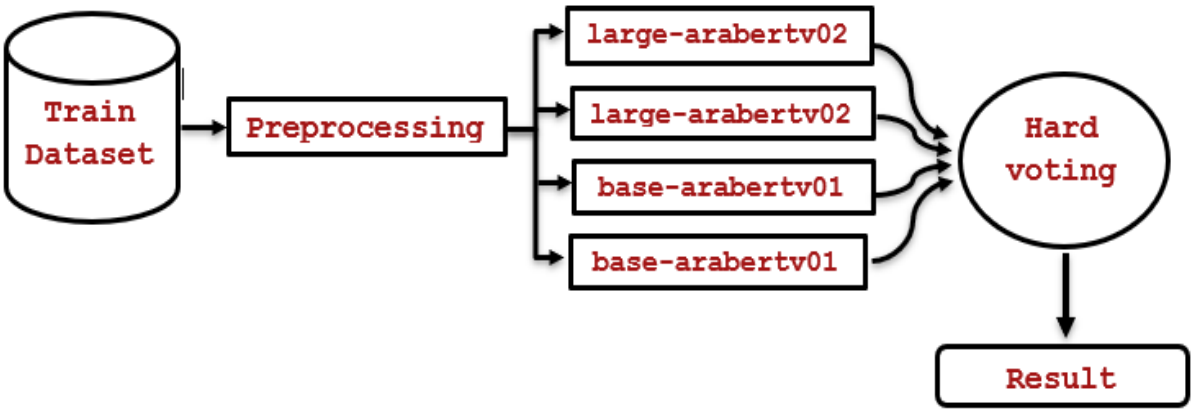

Figure 2: The architecture of our model.

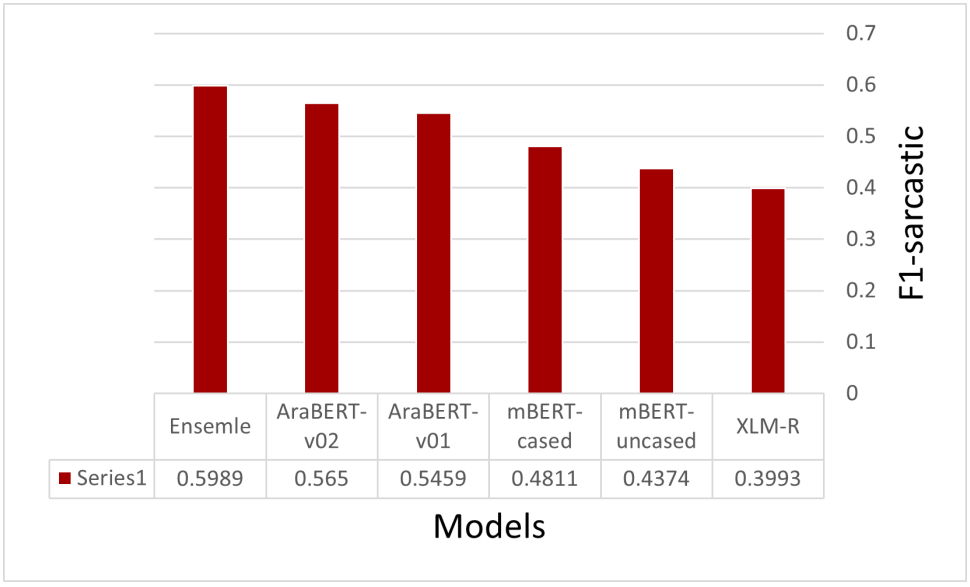

Figure 3: Compresion between result of all models.

#	Model	Epoch	Batch Size	LR	F1-sarcastic
1	AraBERTv01	5	16	1e-5	0.5459
2	AraBERTv02	5	16	1e-5	0.5650
3	XLM-R	5	16	1e-5	0.3993
4	mBERT-cased	5	16	1e-5	0.4811
5	mBERT-uncased	5	16	1e-5	0.4374

Table 3: all the hyper-parameter that we used in our experiments

Rank	F1-sarcastic	Accuracy	Macro F1	Precision	Recall
5th	0.5989	0.7830	0.7251	0.7268	0.7235

Table 4: Official results on subtask 1 (sarcasm Detection) test set.

0.5650 on the development phase. Table 4 shows the organizers' final result and table 5 shows all experiments that we have implemented, and table 6 shows all ensemble results. Finally, we noticed that AraBERT-v02 has strong performance on most complex implementations with the dataset, especially with the hard-voting ensemble technique on imbalanced data classes.

#	Preprocessing	F1-sarcastic
(1)	with	0.3993
(2)	without	0.4374
(3)	with	0.4650
(4)	with	0.4811
(5)	without	0.5419
(6)	with	0.5459
(7)	without	0.5560
(8)	with	0.5650

Table 5: Result for all experiment.

Ensemble	F1-sarcastic
(6)(7)(8)	0.5682
(5)(6)(7)(8)	0.5989

Table 6: All Ensemble result.

7 Conclusion

Our paper showed and described our approach that achieved the best accuracy score model in the Shared Task on Sarcasm and Sentiment Detection in Arabic (Sub-task 1 - Sarcasm Detection). We have implemented and experimented with several NLP-based language models, such as XLM-R, mBERT, and AraBERT, with hard-voting ensemble techniques to detect and solve the sarcasm in Arabic tweets. Our last and best solution is to ensemble the four models scoring F1-sarcastic 0.5650, 0.5459, 0.5560, 0.5419 by applying hard-voting ensemble technique. The SarcasmDet model achieved 0.5989 F1-sarcastic score, which outperformed the baseline model (0.3993 F1-sarcastic). Our proposed model has earned 5th place among 27 teams.

References

Nawaf A Abdulla, Nizar A Ahmed, Mohammed A Shehab, and Mahmoud 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. IEEE.

Malak Abdullah and Mirsad Hadzikadic. 2017. Sentiment analysis on arabic tweets: Challenges to dissecting the language. In *International Conference on Social Computing and Social Media*, pages 191–202. Springer.

Malak Abdullah, Mirsad Hadzikadicy, and Samira Shaikhz. 2018. Sedat: Sentiment and emotion detection in arabic text using cnn-lstm deep learning. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 835–840. IEEE.

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*.

Mahmoud Al-Ayyoub, Abed Allah Khamaiseh, Yaser Jararweh, and Mohammed N Al-Kabi. 2019. A comprehensive survey of arabic sentiment analysis.

Information processing & management, 56(2):320–342.

Dana Al-Ghadhban, Eman Alnkhilan, Lamma Tatwany, and Muna Alrazgan. 2017. Arabic sarcasm detection in twitter. In *2017 International Conference on Engineering & MIS (ICEMIS)*, pages 1–7. IEEE.

Wissam Antoun, Fady Baly, and Hazem Hajj. Arabert: Transformer-based model for arabic language understanding. In *LREC 2020 Workshop Language Resources and Evaluation Conference 11–16 May 2020*, page 9.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Li-Chen Cheng and Song-Lin Tsai. 2019. Deep learning for automated sentiment analysis of social media. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 1001–1004.

Te-Shun Chou, Jeffrey Fan, Sharon Fan, and Kia Makki. 2009. Ensemble of machine learning algorithms for intrusion detection. In *2009 IEEE International Conference on Systems, Man and Cybernetics*, pages 3976–3980. IEEE.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Rehab Duwairi and Mahmoud El-Orfali. 2014. A study of the effects of preprocessing strategies on sentiment analysis for arabic text. *Journal of Information Science*, 40(4):501–513.

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.

Ronen Feldman. 2013. Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4):82–89.

Roberto González-Ibánez, Smaranda Muresan, and Nina Wacholder. 2011. Identifying sarcasm in twitter: a closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–586.

Sabit Hassan, Younes Samih, Hamdy Mubarak, Ahmed Abdelali, Ammar Rashed, and Shammur Absar Chowdhury. 2020. Alt submission for osact shared task on offensive language detection. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 61–65.

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.

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. *arXiv preprint arXiv:1907.11692*.

Kingsley A Ogudo and Dahj Muwawa Jean Nestor. 2019. Sentiment analysis application and natural language processing for mobile network operators' support on social media. In *2019 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*, pages 1–10. IEEE.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.