# Preprocessing Solutions for Detection of Sarcasm and Sentiment for Arabic

**Mohamed Lichouri, Mourad Abbas, Besma Benaziz, Aicha Zitouni, Khaled Lounnas**
Computational Linguistics Department / CRSTDLA, Algiers, Algeria
{m.lichouri, m.abbas, b.benaziz}@crstdla.dz
{a.zitouni, k.lounnas}@crstdla.dz

## Abstract

This paper describes our approach to detecting Sentiment and Sarcasm for Arabic in the Ar-Sarcasm 2021 shared task. Data preprocessing is a crucial task for a successful learning, that is why we applied a set of preprocessing steps to the dataset before training two classifiers, namely Linear Support Vector Classifier (LSVC) and Bidirectional Long Short Term Memory (BiLSTM). The findings show that despite the simplicity of the proposed approach, using the LSVC model with a normalizing Arabic (NA) preprocessing and the BiLSTM architecture with an Embedding layer as input have yielded an encouraging F1score of 33.71% and 57.80% for sarcasm and sentiment detection, respectively.

## 1 Introduction

Sentiment Analysis (SA) is a natural language processing field that aims to detect people's opinions and emotions (Nassif et al., 2020). Recently, SA becomes a big challenge in which several works were realized using different methods. Considering works that investigate twitter/tweets as data source, we can cite (Abdul-Mageed et al., 2019) who describe a collection of deep learning Arabic social media processing tools (AraNet) to analyze 15 datasets related to sentiment analysis of Arabic including MSA and its dialects by the way of a Bidirectional Encoder Representations from Transformers (BERT). (Farha and Magdy, 2020) presented ArSarcasm dataset to train a deep learning model for sarcasm detection based on SA using Bidirectional Long Short Term Memory (BiLSTM). In (Beseiso and Elmousalami, 2020), authors have carried out a comparative study by applying three deep learning techniques: Convolutional Neural Network (CNN), Bidirectional Gated Recurrent Unit (BiGRU), and Attention on two datasets (ASTD and LABR). In the case of a single dialect, we find the work done for the Algerian ALGED dataset by (Moudjari et al., 2020), who used classical and deep learning classification to tackle the problem of sentiment analysis.

One of the main challenges of SA is the Sarcasm detection which could be beneficial in many areas. Sarcasm can be defined as a special form of verbal irony that is intended to express contempt or ridicule (Joshi et al., 2017), where people convey the opposite of what they mean, using implicit indirect phrasing, where the intended meaning is different from the literal one (Wilson, 2006). In recent years, we have noticed that Arabic corpora are diversifying more and more according to the task at hand, for instance, one can mention those built by (Bouamor et al., 2018; Zaghouani and Charfi, 2018; Maamouri et al., 2010; Zaghouani et al., 2014; Bouamor et al., 2015). In addition, irony and sarcasm detection has recently drawn a significant attention in computational linguistics (Joshi et al., 2017). Fewer studies considered in detail irony detection in Arabic. The only and earliest corpus on Arabic sarcasm/irony detection is SOUKHRIA corpus in (Karoui et al., 2017), where the authors created a corpus of Arabic tweets, by collecting a set of political keywords. They used the Arabic equivalent of sarcasm # , # #. However, this corpus has not been released to public yet. There is also the work done by (Al-Ghadhban et al., 2017) who proposed a classification model that detects Arabic-sarcasm tweets by using some data mining algorithms. In the first shared task on irony detection for the Arabic language organized by (Ghanem et al., 2019), where they collected their data using distant supervision and used similar Arabic hashtags. The task consists of a binary classification of tweets as ironic or not using a dataset composed of 5,030 Arabic tweets about different political issues and events related to the Middle East and the Maghreb. Another contribution to the creation of



*Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 376–380
Kyiv, Ukraine (Virtual), April 19, 2021.

| | Train (ArSarcasm-v2) | Dev | Test (ArSarcasm-v2) |
|---|---|---|---|
| # sentences | 12,548 | 2,110 | 3,000 |
| # words | 275,854 | 46,456 | 64,347 |
| Max # word per sentence | 148 | 52 | 92 |
| Min # word per sentence | 1 | 1 | 1 |
| Max # char per sentence | 298 | 52 | 92 |
| Min # char per sentence | 2 | 1 | 1 |

Table 1: ArSarcasm-v2 dataset statistics

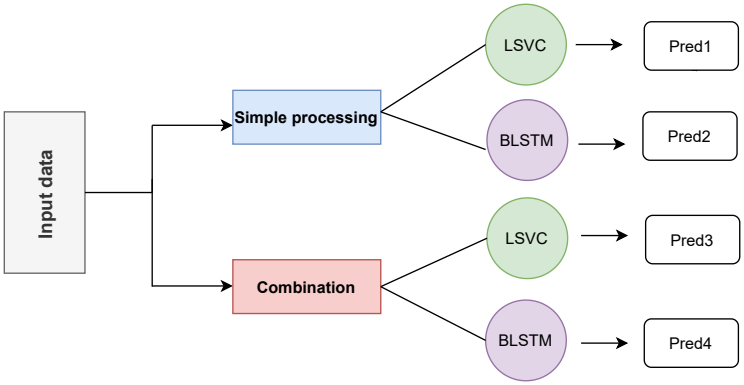


Figure 1: Sarcasm and sentiment detection system.

new corpus, (Abbes et al., 2020) proposed a new open domain Arabic corpus annotated for irony detection, which was also collected from Twitter. The rest of the paper is organized as follows: In section 2, a description of the used dataset is presented. The applied cleaning steps and preprocessing are explained in section 3. In section 4, we expose the proposed approach and experiments. Finally, the findings and discussion are presented in section 5. We conclude the paper in section 6.

## 2 Description of the Dataset

ArSarcasm-v2 dataset (Abu Farha et al., 2021) has been built using previously available Arabic sentiment analysis datasets (SemEval 2017 (Rosenthal et al., 2017) and ASTD (Nabil et al., 2015)) for which sarcasm and dialect labels have been assigned to them. More details about this dataset is addressed, in Table 1. Note that these statistics are related to the dataset after removal of punctuation and emojis.

## 3 Data Cleaning and Preprocessing

The cleaning process is the first step to apply with Arabic tweets. We define it as a surface preprocessing which includes one or many of the following steps: punctuation removal, emojis removal, stop words removal, Arabic diacritics removal, Arabic Letter normalization, Latin letter and words removal, repeating words and chars removal. The second step is morphological preprocessing: lemmatization (WorADRetLemmatizer (Lem)), stemming (ISRI Arabic Stemmer (Stem)) and part of speech tagging (PosTagger (PosTag) of NLTK)[1] (Lichouri and Abbas, 2020; Lichouri et al., 2020).

## 4 Experiments

As our focus is on detecting sentiment and sarcasm, we present, in this section, our contribution which is preprocessing as a solution to improve detection, in order to build LSVC and BiLSTM classification models (see figure 1). After performing this preprocessing step, features are calculated using the TFIDF vectorizer. These features are used to learn the LSVC model, where default parameters defined in the sklearn library are used (Pedregosa et al., 2011).

For BiLSTM, we adopted RNN model which uses an embedding layer with an input of 20k words which will be converted to vector with size 50. We fixed the max length of the input sequences (tweets) to 70 words (Table 1, max token length in train is 148 and test 92). After that we added a Bidirectional LSTM layer with 512 units, followed by a max pooling layer. We then added a dense layer

[1]https://www.nltk.org/index.html

| | Sentiment | | Sarcasm | |
|---|---|---|---|---|
| Params\Models | BiLSTM | LSVC | BiLSTM | LSVC |
| Default (Without Pre-processing) | 88.23 | 98.21 | **86.05** | 98.83 |
| Arabic Diacretics Removal (ADR) | 86.66 | 98.23 | 57.76 | 98.68 |
| normalizeArabic (NA) | 88.29 | **98.46** | 83.87 | 98.83 |
| remove_emoji (RE) | 86.04 | 98.32 | 75.61 | 98.83 |
| remove_repeating_char (RRC) | 86.83 | 98.23 | 85.98 | 98.83 |
| removeLatinLetter (RLL) | 87.20 | 98.26 | 79.72 | 98.68 |
| removeOneLetterWord (ROLW) | 86.25 | 98.21 | 84.51 | 98.83 |
| removePunctuation (RP) | 87.34 | 98.35 | 84.67 | 98.68 |
| removeStopWord (RSW) | 86.81 | 98.11 | 84.89 | 98.83 |
| removeWordRepetition (RWR) | 86.64 | 98.33 | 81.47 | 98.83 |
| applyLemme (Lemme) | 86.31 | 98.21 | 81.63 | 98.83 |
| applyPosTag (PosTag) | 52.77 | 73.39 | 32.38 | 98.83 |
| applyStem (Stem) | 87.61 | 98.32 | 82.1 | 98.83 |

Table 2: Obtained score in the development phase in both sarcasm (F1-sarcastic) and sentiment (F1-PN) detection using a combination of processing steps.

| | Sentiment | | Sarcasm | |
|---|---|---|---|---|
| Params\Models | BiLSTM | LSVC | BiLSTM | LSVC |
| RP+RE | 85.95 | 98.33 | 85.76 | 98.68 |
| RP+RE+RSW | 85.53 | 98.14 | 70.73 | 98.53 |
| RP+RE+RSW+RRC | 86.77 | 98.26 | 85.14 | 98.53 |
| RP+RE+RSW+RRC+NA | 87.08 | 98.22 | 85.22 | 98.68 |
| RP+RE+RSW+RRC+NA +ADR | 86.79 | 98.22 | 81.32 | 98.53 |
| RP+RE+RSW+RRC+NA +ADR+RWR | 87.74 | 98.23 | 84.58 | 98.53 |
| RP+RE+RSW+RRC+NA +ADR+RWR+ROLW | 87.44 | 98.23 | 75.56 | 98.53 |
| RP+RE+RSW+RRC+NA +ADR+RWR+ROLW+RLL | 87.17 | 97.62 | 84.62 | 97.94 |
| RP+RE+RSW+RRC+NA +ADR+RWR+ROLW+RLL+Stem | 86.80 | 97.08 | 82.57 | 97.03 |
| RP+RE+RSW+RRC+NA +ADR+RWR+ROLW+RLL +Stem+Lemme | 87.25 | 97.46 | 80.88 | 98.09 |
| RP+RE+RSW+RRC+NA +ADR+RWR+ROLW+RLL +Stem+Lemme+PosTag | 66.63 | 95.42 | 56.47 | 91.22 |

Table 3: Obtained score in the development phase in both sarcasm (F1-sarcastic) and sentiment (F1-PN) detection using a combination of processing steps with LSVC and BiLSTM.

of 256 units followed by a dropout layer of 0.4, and a second dense layer with 2 units (Sentiment: 2 classes) or 3 units (Sarcasm: 3 classes). The BiLSTM model is compiled using the binary and categorical cross entropy and the RMSprop for optimization. For training, we used a batch size of 128, 5 epochs, and a validation split of 0.2. For development, we used the test set from the ArSarcasm corpus [2]. The results are reported in Tables 2 and 3.

[2]https://github.com/iabufarha/ArSarcasm

| | Sarcasme Task | | Sentiment Task | |
|---|---|---|---|---|
| | F1-sarcastic | Accuracy | F1-PN | Accuracy |
| Our proposed system | 33.71% | 72.87% | 57.87% | 59.23% |
| Average for all system | 52.52% | 80.48% | 65.43% | 64.61% |

Table 4: Obtained results of our submitted system vs the average for all the participant for sarcasm and sentiment detection in the test phase.

## 5 Results and Discussion

In Table 2, we summarize the results obtained using the development set from Abu Farha GitHub. All the preprocessing and morphological processing steps are applied independently.

The reported results shows that in the case of sentiment detection, the best performance is obtained using the Arabic normalizer with an F-PN score of 88.29% and 98.46% with BiLSTM and LSVC, respectively. In the case of sarcasm detection, the best performance is achieved without applying any preprocessing step with an F1-sarcastic score of 86.05% and 98.83% with BiLSTM and LSVC, respectively. We should note that the combination of preprocessing steps didn't improve the performance of the system (Table 3).

### Results on the test set

Our submitted prediction during the test phase is based on the models that use preprocessing and morphological processing steps independently. The performance we achieved for the test phase was F1-sarcastic = 33.71% and F-PN = 57.87% which is less than the average values of all submitted systems by around (+8%) and (+7%) for sarcasm and sentiment detection, respectively (see Table 4).

## 6 Conclusion

In this work, we presented a simple but intuitive detection system based on the investigation of a number of preprocessing steps and their combinations. A comparison between LSVC and BiLSTM classfiers was conducted where we tried to find the best combination of "preprocessing + classifiers". After conducting more than 200 experiments, we found that feeding BiLSTM (used for sarcasm detection) with raw text without preprocessing is better and allowed to achieve a score of 33.71%. In the case of LSVC (used to detect the sentiment of tweets), we found that the better preprocessing step, in our case, is the Arabic Letter Normalizer with an achieved score of 57.87%.

## References


Ines Abbes, Wajdi Zaghouani, Omaima El-Hardlo, and Faten Ashour. 2020. Daict: A dialectal arabic irony corpus extracted from twitter. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6265–6271.

Muhammad Abdul-Mageed, Chiyu Zhang, Azadeh Hashemi, and El Moatez Billah Nagoudi. 2019. Aranet: A deep learning toolkit for arabic social media. *arXiv preprint arXiv:1912.13072*.

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*.

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.

Majdi Beseiso and Haytham Elmousalami. 2020. Subword attentive model for arabic sentiment analysis: A deep learning approach. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(2):1–17.

Houda Bouamor, Nizar Habash, Mohammad Salameh, Wajdi Zaghouani, Owen Rambow, Dana Abdulrahim, Ossama Obeid, Salam Khalifa, Fadhl Eryani, Alexander Erdmann, et al. 2018. The madar arabic dialect corpus and lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Houda Bouamor, Wajdi Zaghouani, Mona Diab, Ossama Obeid, Kemal Oflazer, Mahmoud Ghoneim, and Abdelati Hawwari. 2015. A pilot study on arabic multi-genre corpus diacritization. In *Proceedings of the Second Workshop on Arabic Natural Language Processing*, pages 80–88.

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.

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.

Aditya Joshi, Pushpak Bhattacharyya, and Mark J Carman. 2017. Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50(5):1–22.

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.

Mohamed Lichouri and Mourad Abbas. 2020. Speechtrans@ smm4h'20: Impact of preprocessing and n-grams on automatic classification of tweets that mention medications. In *Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task*, pages 118–120.

Mohamed Lichouri, Mourad Abbas, and Besma Benaziz. 2020. Profiling fake news spreaders on twitter based on tfidf features and morphological process.

Mohamed Maamouri, Ann Bies, Seth Kulick, Wajdi Zaghouani, David Graff, and Michael Ciul. 2010. From speech to trees: Applying treebank annotation to arabic broadcast news. In *LREC*. Citeseer.

Leila Moudjari, Karima Akli-Astouati, and Farah Benamara. 2020. An algerian corpus and an annotation platform for opinion and emotion analysis. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 1202–1210.

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.

Ali Bou Nassif, Ashraf Elnagar, Ismail Shahin, and Safaa Henno. 2020. Deep learning for arabic subjective sentiment analysis: Challenges and research opportunities. *Applied Soft Computing*, page 106836.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

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.

Deirdre Wilson. 2006. The pragmatics of verbal irony: Echo or pretence? *Lingua*, 116(10):1722–1743.

Wajdi Zaghouani and Anis Charfi. 2018. Arap-tweet: A large multi-dialect twitter corpus for gender, age and language variety identification. *arXiv preprint arXiv:1808.07674*.

Wajdi Zaghouani, Behrang Mohit, Nizar Habash, Ossama Obeid, Nadi Tomeh, Alla Rozovskaya, Noura Farra, Sarah Alkuhlani, and Kemal Oflazer. 2014. Large scale arabic error annotation: Guidelines and framework.