High Tech team at SemEval-2022 Task 6: Intended Sarcasm Detection for Arabic texts

Alami Hamza[1], **Abdessamad Benlahbib**[2], **Ahmed Alami**[3],
[1] Innov-Tech Laboratory, Departement of Engineering, High Technology School, Rabat, Morocco
[2] LISAC Laboratory, Faculty of Sciences Dhar EL Mehraz (F.S.D.M),
Sidi Mohamed Ben Abdellah University (U.S.M.B.A)
[3] Ibn Tofail University, National School of Applied Sciences, Kenitra, Morocco
hamza0alami@gmail.com, abdessamad.benlahbib@usmba.ac.ma,
alami.alami1996@gmail.com

Abstract

This paper presents our proposed methods for iSarcasmEval shared task. The shared task consists of three different subtasks. We participate in both subtask A and subtask C. The purpose of the subtask A was to predict if a text is sarcastic while the aim of subtask C is to determine which text is sarcastic given a sarcastic text and its non-sarcastic rephrase. Both of the developed solutions used BERT pre-trained models. The proposed models are optimized on simple objectives and easy to grasp. However, despite their simplicity our methods ranked 4 and 2 in iSarcasmEval subtask A and subtask C for Arabic texts.

1 Introduction

Nowadays, social media users provide a huge amount of text, images and videos. This large amount of data contains useful information (users ideas, opinions, events, etc) for various domains such as stock predictions, marketing, or politics. In order to benefit from these data, new fields of study have been introduced including sentiment analysis, opinion mining, author profiling, and harassment detection (Liu, 2012; Rosenthal et al., 2014; Maynard and Greenwood, 2014; Van Hee et al., 2018). Natural Language Processing (NLP) algorithms are used extensively in these fields to extract useful information. For instance, to determine whether a given product has a positive or negative sentiment in the market, we can apply NLP techniques to analyse a list of twitter posts to infer a sentiment about the product.

According to Oxford dictionary, sarcasm is "*The use of irony to mock or convey centempt*". Sarcastic text convey negative implied sentiment, however it can have positive, negative, or no surface sentiment. Sarcasm is commonly used in social media, thus it introduces errors in various tasks such as sentiment analysis and opinion mining. This is explained in the work of Rosenthal et al. (2014), it shows a significant drop in sentiment polarity classification performance when processing sarcastic tweets, compared to non-sarcastic ones. In this context, the task *iSarcasmEval: Intended Sarcasm Detection In English and Arabic* (Abu Farha et al., 2022) is organized by SemEval 2022. The main tasks consists of three subtask:

- Subtask A: Given a text, determine whether it is sarcastic or non-sarcastic.
- SubTask B (English only): A binary multi-label classification task. Given a text, determine which ironic speech category it belongs to, if any.
- SubTask C: Given a sarcastic text and its non-sarcastic rephrase, i.e. two texts that convey the same meaning, determine which is the sarcastic one.

In this paper, we describe our contribution to iSarcasmEval shared task, Arabic language only. For subtask A, we built a BERT-based neural network (Devlin et al., 2019; Antoun et al., 2020) classifier to determine whether a tweet is sarcastic or not.

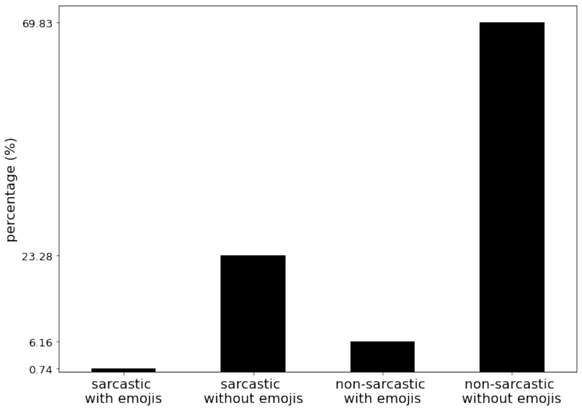

Figure 1: The train dataset distribution according to 4 classes including sarcastic texts that contains emojis, sarcastic texts without emojis, non-sarcastic texts with emojis, and non-sarcastic texts without emoji.

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 840 - 843
July 14-15, 2022 ©2022 Association for Computational Linguistics

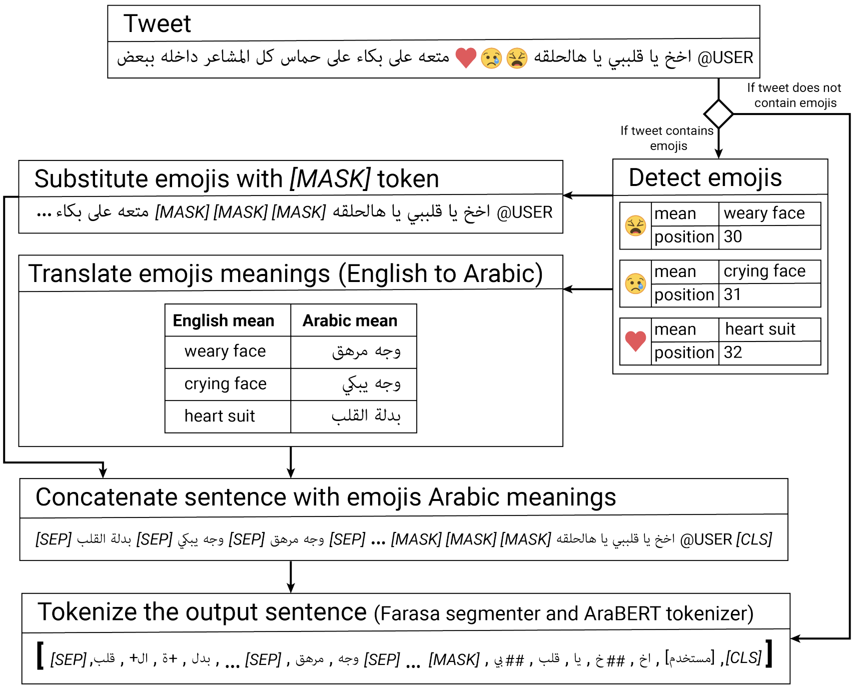
Tweet
@USER اخخ يا قلبي يا هالحلقه 😫😢❤️ متعه على بكاء على حماس كل المشاعر داخله ببعض
If tweet does not contain emojis
If tweet contains emojis
Detect emojis
mean | weary face
position | 30
mean | crying face
position | 31
mean | heart suit
position | 32
Substitute emojis with [MASK] token
@USER اخخ يا قلبي يا هالحلقه [MASK] [MASK] [MASK] متعه على بكاء ...
Translate emojis meanings (English to Arabic)
English mean | Arabic mean
weary face | وجه مرهق
crying face | وجه يبكي
heart suit | بدلة القلب
Concatenate sentence with emojis Arabic meanings
[CLS] @USER اخخ يا قلبي يا هالحلقه [MASK] [MASK] [MASK] ... [SEP] وجه مرهق [SEP] وجه يبكي [SEP] بدلة القلب [SEP]
Tokenize the output sentence (Farasa segmenter and AraBERT tokenizer)
[[CLS], [مستخدم] , اخ , ##خ , يا , قلب , ##ي , [MASK] ... [SEP] وجه , مرهق , [SEP] ... , بدل , +ة , ال+ , قلب, [SEP]]

Figure 2: Text preprocessing proposed by Alami et al. (2020).

Our model obtained the fourth best performance in the subtask A. For subtask C, we built also a BERT-based classifier to detect the sarcastic text from two text that convey the same meaning. We scored the second best performance in the subtask C. The results are promising and there is much room for improvement.

The rest of the paper is organized as follows: Section 2 presents our method overview; Section 3 provides performance evaluation; Section 4 concludes the paper and provides future work.

2 Method Overview

In this section, we first describe how we split data to evaluate our models. Next, we explain preprocessing steps. Next, we discuss our models for each subtask, and the experimental setup we used. We also provide illustrations and examples, when necessary.

2.1 Dataset split

The organizers of iSarcasmEval provided Arabic texts annotated with their sarcasm labels. The train set contains 3102 samples where 75.98% (2357 samples) are non-sarcastic and 24.02% (745 samples) are sarcastic. The test set consists of 1400 samples. All the samples are annotated also with their dialect. We build a validation set from train set based on emojis. We first split the train data into 4 classes including sarcastic texts that contains emojis, sarcastic texts without emojis, non-sarcastic texts with emojis, and non-sarcastic texts without emojis. Fig. 1 illustrates the distribution of this 4 classes in the train dataset. We notice that only 6.9% of train samples contain emojis while 20.86% test samples include emojis. Considering this we use 4 splits to validate our models:

- Split A: The validation set contains all the sarcastic samples with emojis, 10% sarcastic samples without emojis, 10% non-sarcastic samples with emojis, and 10% non-sarcastic samples without emojis.
- Split B: The validation set contains 50% sarcastic samples with emojis, 10% sarcastic samples without emojis, 10% non-sarcastic samples with emojis, and 10% non-sarcastic samples without emojis.
- Split C: The validation set does not contain any sarcastic samples with emojis and contains 10% sarcastic samples without emojis, 10% non-sarcastic samples with emojis, and 10% non-sarcastic samples without emojis.
- Split D: The validation set contains 20% of train samples. We applied stratified split to have the same distribution of classes as the train set.

Table 1: Performance evaluation of different models for sarcasm prediction

	Split A			**Split B**			**Split C**			**Split D**			**Score**
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	
AraBERTv02-twitter	84.76	64.10	72.99	76.92	73.53	75.19	75.00	82.76	78.69	78.66	86.58	82.43	**77.32**
CAMEL-MIX	86.67	66.67	75.36	78.95	66.18	72.00	74.24	84.48	79.03	77.07	81.21	79.08	76.36
AraBERTv02-twitter / Emojis	90.74	62.82	74.24	55.93	48.53	51.97	75.41	79.31	77.31	77.22	81.88	79.48	70.75

2.2 Preprocessing

Our preprocessing step consists of tokenization. We apply the pre-trained BERT tokenizer which is based on wordpiece model (Schuster and Nakajima, 2012). For comparison purposes, we applied the same preprocessing process applied by Alami et al. (2020). The main idea is to integrate the meaning of emojis whitin the initial text. Fig. 2 presents the preprocessing step used in (Alami et al., 2020).

2.3 SubTask A

The objective of this task is to predict whether a text is sarcastic or not. We fin tune various BERT-based models pre-trained with Arabic large corpora. These models are used to extract valuable features from raw text. These features are then used with a softmax classifier to predict the label of the input text. All models are optimized to minimize the cross entropy loss.

2.4 SubTask C

The aim of this task is to predict the sarcastic text given two texts with the same meaning. Like the model used in subtask A, we fine tune BERT-based models for this specific task. The input of these models is the concatenation of the two texts separated by the special token *[SEP]*. Features are extracted with BERT-based models, then a softmax layer is applied to compute the probabilities of the events: first text is sarcastic and second text is sarcastic. All models are optimized to minimize the cross entropy loss.

2.5 Experimental Setup

We implemented our models using HuggingFace (Wolf et al., 2020). We used AraBERTv02-twitter (Antoun et al., 2020) and CAMEL-Mix (Inoue et al., 2021) as the pre-trained language models. To train our models, we used a batch size of 8, a learning rate 10^{-5}. We used the AdamW optimizer (Loshchilov and Hutter, 2017). We ran the experiments on a Google colaboratory environment [1].

3 Performance Evaluation

In this section, we present the performance of various models trained on both subtasks A and C.

3.1 Subtask A

First, we compared the performances of two models: a model fine tuned with AraBERTv02-twitter and a model fine tuned with CAMEL-MIX. Table 1 shows the obtained results according to the 4 splits we previously discussed in subsection 2.1. We compute the overall score of a model by averaging all the f1 scores of the sarcastic class obtained from different splits. The model fine tuned with AraBERTv02-twitter scored the best results.

To investigate the impact of the substitution of emojis with their meanings. We evaluated the performances of a model based on AraBERTv02-twitter and take as input text preprocessed as proposed in Alami et al. (2020). Table 1 shows that emojis processing didn't improve the overall score.

Therefore, we submit the predictions obtained using AraBERTv02-twitter with the test set. We scored the fourth best score in the leaderboard (46.84% f1 score for sarcastic class).

3.2 Subtask C

Since the AraBERTv02-twitter model obtained the best result in subtask A, we trained the same base model to predict the sarcastic text given two text that convey the same meaning. We augmented the dataset by applying a simple rule which consist of switching the positions of the sarcastic text with the non-sarcastic text and replace the label with 0. We ranked 2 in the leaderboard with (88.5% accuracy).

4 Conclusion

We developed two methods for sarcasm prediction. The first one aim to predict if a text is sarcastic or not. This method is based on AraBERTv02-twitter pretrained model which extract valuable features from raw text. We achieved the fourth top performance in the iSarcasmEval subtask A for Arabic with a 46.84% f1 score for the sarcastic class. The second model has the objective to detect

[1] https://colab.research.google.com/

the sarcastic text given two texts that convey the same meaning. We trained a BERT-based model that take as input two text and predict the sarcastic one. We ranked 2 in the leaderboard with 88.5% accuracy. In future work, we plan to improve the performance of our models by using linguistic rules and some external datasets.

References

Ibrahim Abu Farha, Silviu 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)*. Association for Computational Linguistics.

Hamza Alami, Said Ouatik El Alaoui, Abdessamad Benlahbib, and Noureddine En-nahnahi. 2020. LISAC FSDM-USMBA team at SemEval-2020 task 12: Overcoming AraBERT's pretrain-finetune discrepancy for Arabic offensive language identification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 2080–2085, Barcelona (online). International Committee 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.

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.

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*, Kyiv, Ukraine (Online). Association for Computational Linguistics.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *ArXiv*, abs/1711.05101.

Diana Maynard and Mark Greenwood. 2014. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4238–4243, Reykjavik, Iceland. European Language Resources Association (ELRA).

Sara Rosenthal, Alan Ritter, Preslav Nakov, and Veselin Stoyanov. 2014. SemEval-2014 task 9: Sentiment analysis in Twitter. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 73–80, Dublin, Ireland. Association for Computational Linguistics.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152.

Cynthia Van Hee, Gilles Jacobs, Chris Emmery, Bart Desmet, Els Lefever, Ben Verhoeven, Guy De Pauw, Walter Daelemans, and Véronique Hoste. 2018. Automatic detection of cyberbullying in social media text. *PLOS ONE*, 13(10):1–22.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.