Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, pages 1–8
with a Shared Task on Offensive Language Detection.
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

An Arabic Tweets Sentiment Analysis Dataset (ATSAD) using Distant Supervision and Self Training

Kathrein Abu Kwaik[*], Motaz Saad[†], Stergios Chatzikyriakidis[*], Simon Dobnik[*], Richard Johansson[¶]
[*]CLASP, Department of Philosophy, Linguistics and Theory of Science, University of Gothenburg, Sweden
[¶]Department of Computer Science and Engineering, University of Gothenburg, Sweden
[†]The Islamic University of Gaza, Palestine
{kathrein.abu.kwaik,richard.johansson,stergios.chatzikyriakidis,simon.dobnik}@gu.se , motaz.saad@gmail.com

Abstract

As the number of social media users increases, they express their thoughts, needs, socialise and publish their opinions. For good social media sentiment analysis, good quality resources are needed, and the lack of these resources is particularly evident for languages other than English, in particular Arabic. The available Arabic resources lack of from either the size of the corpus or the quality of the annotation. In this paper, we present an Arabic Sentiment Analysis Corpus collected from Twitter, which contains 36K tweets labelled into positive and negative. We employed distant supervision and self-training approaches into the corpus to annotate it. Besides, we release an 8K tweets manually annotated as a gold standard. We evaluated the corpus intrinsically by comparing it to human classification and pre-trained sentiment analysis models. Moreover, we apply extrinsic evaluation methods exploiting sentiment analysis task and achieve an accuracy of 86%.

Keywords: Sentiment Analysis, Distant Supervision, Self Training

1. Introduction

Companies and businesses stakeholders reach out to their customers through Social Media not only for advertising and marketing purposes, but also to get customer feedback concerning products or services. This is one of the main reasons that sentiment analysis applications have become increasingly sought out by the industry field. Even though sentiment analysis programs are widely used in the commercial sector, they have many other important uses, including political orientation analysis, electoral programs and decision-making. Sentiment Analysis is the process of automatically mining attitudes, opinions, views and emotions from the text, speech, tweets using Natural Language Processing (NLP) and machine learning (Liu, 2012). Sentiment analysis involves classifying opinions into different classes like positive, negative, mixed or neutral. It can also refer to Subjectivity Analysis, i.e. the task of distinguishing between objective and subjective text.

There are so many Arabic speakers in the world and they speak different varieties of Arabic depending on the region but with only one variety that is standardised namely, Modern Standard Arabic MSA. Social media is prevalent and it is particularly this domain where the local varieties are used and for which the resources are most limited. The total number of monthly active Twitter users in the Arab region is estimated at 11.1 million in March 2017, generating 27.4 million tweets per day according to weedoo.[1] Arabic, especially dialects, still looking for more efficient resources that can be used for the needs of NLP tasks.

One of the biggest challenges in the construction of Arabic NLP resources is the big variation found in Arabic language where there are Modern Standard Arabic (MSA), Classical Arabic (CA) and the dialects. This has the result, that, in some tasks, it might be necessary to build stand-alone resources for each individual variation where the available tools have been built for MSA can not be adapted for dialects and vice-verse (Qwaider et al., 2019). In addition, building resources requires sufficient time and funding to produce highly efficient resources. Moreover, deep learning NLP methods require a huge amount of data. As a result of the unique Twitter features that are widely used to express opinions, views, thoughts and feelings, we therefore present Arabic Tweets Sentiment Analysis Dataset (ATSAD) contains 36k tweets classified as positive or negative.

The contributions of this paper can be highlighted under two headings: a) resource creation and b) resource evaluation. Regarding resource creation, we introduce a sentiment analysis dataset collected from Twitter, and as for resource evaluation, we introduce a method that combines the distant supervision approach with self-training to build a dataset that satisfies the size and quality requirements. In order to annotate a large number of tweets, we employ the distant supervision approach where the emojis are used as a weak noisy label. We manually annotate a subset of 8k tweets of the dataset and offer it as gold standard dataset. In order to improve the quality of the corpus, we apply the self-training techniques on the dataset and combine it with the distant supervision approach as a *double check approach.* Using our proposed double check approach, we achieve an accuracy of 86% on the sentiment analysis task. The dataset is available online for research usage.[2]

The rest paper is organised as follows: Section 2 reviews some related works in term of sentiment analysis and social media resources. In Section 3, the challenges of processing Twitter text are presented and in Section 4, the details of collecting and creating the tweets dataset are presented. We evaluate the dataset in Section 5. Sections 6 and 7 are the conclusion and future work sections respectively.

[1] https://weedoo.tech/twitter-arab-world-statistics-feb-2017/

[2] https://github.com/motazsaad/arabic-sentiment-analysis

2. Related Work

Arabic Sentiment analysis (ASA) has received considerable attention in terms of resource creation (Rushdi-Saleh et al., 2011; Aly and Atiya, 2013; Abdul-Mageed et al., 2014; Elnagar et al., 2018). These resources are collected from different sources such as (blogs, reviews, tweets, comments, etc.) and involve a mix of Arabic vernacular and classical Arabic. Furthermore, they have been used extensively in research on SA for Arabic such as (Al Shboul et al., 2015; Obaidat et al., 2015; Al-Twairesh et al., 2016). Most NLP work on SA uses machine learning classifiers with feature engineering. For example (Azmi and Alzanin, 2014; El-Beltagy et al., 2016) used machine learning classifiers on polarity and subjectivity classifications. However, recent papers (Al Sallab et al., 2015; Dahou et al., 2016; Alayba et al., 2018) investigated the use of Deep Neural Networks for Arabic sentiment analysis. Most of the datasets are collected from web blogs and customer reviews. Some are manually annotated following a specific annotation guidelines, while other corpora like LABR (Aly and Atiya, 2013) depend on the stars ratings done by users where the stars are used as polarity labels, the 5 stars denote a high positive, 1 star denotes a high negative and the 3 stars indicate the neutral and mixed label.

In the AraSenTi-tweets corpus (Al-Twairesh et al., 2017), many approaches to collect the tweets were adopted, e.g the utilisation of emoticons, sentiment hashtags as well as the sentiment keywords. Then, the authors only keep the tweets that have their location set to a Saudi location. The dataset is manually annotated and sets some annotation guidelines. It contains 17 573 tweets each of which is classified to one of four classes (positive, negative, mixed or neutral). A sentiment baseline is built depending on TFIDF and using SVM with a linear kernel which achieved an F-score of 60.05%.

In (Nabil et al., 2015), the authors presented the Arabic Sentiment Tweets Dataset (ASTD). It is a dataset of 10,000 Egyptian tweets. It is composed of 799 positive, 1,684 negative, 832 mixed and 6,691 neutral tweets. The authors also conducted a set of benchmark experiments for four way sentiment classification as (positive, negative, mixed, neutral) and two-way sentiment classification as (positive, negative). When focusing on two-way classification, the corpus is unbalanced and small to be useful for the two-way sentiment analysis task.

A corpus for Jordanian tweets is also presented in (Atoum and Nouman, 2019). The authors collected tweets according to location, and then they filtered them to collect different types of terminologies to identify Jordanian Arabic dialect keywords efficiently. The corpus contains 3,550 Jordanian dialect tweets manually annotated as follows: 616 positive tweets, 1,313 negative tweets, and 1,621 neutral tweets. They conducted several experiments both with and without stemming/rooting applying them to several models with uni-grams/bi-grams and trying NB and SVM classifiers. The result shows that the SVM classifier performs better than the NB classifier. The ROC performance reached an average of 0.71, 0.77 on NB and SVM respectively on all experiments. A similar corpus for Levantine dialects is presented in Shami-Senti (Qwaider et al., 2019). It has approximately 2.5k posts from social media sites in general topics classified manually as positive, negative and neutral. The corpus is still under development.

Recently, a 40K tweets dataset is presented in (Mohammed and Kora, 2019). The authors extracted tweets written in Arabic. After that, they reprocessed the tweets and cleaned them very carefully by two experts, they corrected every misspelling words and removed all the repeating characters, in addition to the normal cleaning steps like normalisation. The total size of the dataset is 40,000 tweets classified into positive and negative equally. The corpus is considered a reliable resource but by manually cleaning all the data, it turns to a very hard crafted corpus where the resulted clean corpus differ than the real tweets, where the goal of cleaning is to normalise text and remove spelling mistakes but keep the style of the author. This has been normalised too much in this corpus and hence important information was lost.

Even though in most of the Arabic tweet corpus creation procedures, the authors used the emoticons to extract as many sentiment tweets as possible such as (Al-Twairesh et al., 2017; Refaee and Rieser, 2014), however none of them using the emojis and the emoticons as a sentiment label. An emoticon is built from keyboard characters that when put together in a certain way represent a facial expression like :) ;) :(and so on, while an emoji is an actual image[3]. The Stanford Twitter Sentiment (STS), is one of the most well-known dataset for English Twitter sentiment analysis (Go et al., 2009). The dataset provides training and testing sets. The tweets were collected on the condition to contain at least one emoticon. Then they automatically classified the tweets in regard to the emoticons to positive and negative. The process resulted in a training set of 1.6 million annotated tweets and a test set of 359 manually annotated tweets that are used as a gold standard. The data set has been extensively used for different tasks related to sentiment analysis and subjectivity classification (Bravo-Marquez et al., 2013; Saif et al., 2012; Bakliwal et al., 2012; Speriosu et al., 2011). Refaee and Rieser (2014) presented Arabic subsets of tweets using emoticons, hashtags and keywords. They apply distant supervision on the emoticons subset. After the evaluation process, they get an accuracy 95% and 51% for subjectivity analysis and sentiment classification respectively. They comment that emoticons can be used efficiently with subjectivity detection but not for the polarity classification task.

As obvious from the previous discussion, these corpora or dataset have lacked some aspect. They have some limitation in term of the size of the corpus as ASTD, the number of presented dialects as AraSenti and the annotation procedure like LABR. We are looking for Arabic sentiment analysis corpus that concerns the Arabic social media text and that handles multiple dialects in a reasonable number of instances size to conduct experiments and find a way to do the annotation as accurate as possible. In this paper, and similarly to STS (Go et al., 2009), we constructed a dataset based on emojis for extracting and classifying tweets. Additionally, we manually annotated 20% of this data, which

[3] https://grammarist.com/new-words/emoji-vs-emoticon/

can then be used as a gold standard for any tweets sentiment analysis task and as the test set for our corpus.

3. Challenges of processing text from social media

Natural language processing must be adapted to the type of text to be processed (formal, scientific, colloquial), but furthermore, humans differ in the way they write in that specific type of text. This variety in writing style has increased with the advent of social media, where people are using their style of writing and daily conversational language to post, reply, or tweet more often. In addition to specific idiosyncrasies of Arabic in terms of processing, Twitter has unique features that make tweets have different characteristics from other social media (Alwakid et al., 2017; Giachanou and Crestani, 2016). Detecting sentiment in social media text in general and Twitter in particular is a non-trivial task. There are many challenges as follows:

- The short text length is the unique characteristics of tweets, which can be up to 280 characters.
- Due to the constraint on the length of the tweet (280 characters), users tend to employ abbreviations in the tweets to make room for other words.
- Tweets, as well as other social media text, are an example of *User Generated Content*, and contain unstructured language, orthographic mistakes, use of slang words, a lot of ironic and sarcastic sentences, abbreviations and many idiomatic expressions.
- Analysing Arabic tweets in specific is a challenging task due to the use of Arabic dialects in tweets which (due to the lack of standard orthography) results to a lot of spelling inconsistencies. Moreover, the lack of capitalisation and diacritics, as well as the usage of connected words like إنشالله increase the complexity of processing Arabic tweets.
- The extensive of use of misspellings Arabic result in a Data Sparsity, that has an impact on the overall performance of SA systems. Saif et al. (2012) propose a semantic smoothing model by extracting semantically hidden concepts from tweets and then incorporate them into supervised classifier training through interpolation to reduce the sparseness in English tweets.
- Many Arabic tweets are verses from the Holy Quran. There prayers to refer to different situations with different meanings are used, for example, ماما بشتاقلك كتير. الله يرحمك ويجمعنا معك في الجنة, which in English means *Mam I miss you a lot. I ask God to have mercy on you and to bring us together in heaven*, even though it ostensibly carries a positive meaning of empathy and paradise, it carries negative feelings of longing and loss due to death.

4. Arabic Tweets Sentiment Analysis Dataset (ATSAD)

To create and build the sentiment analysis corpus or datasets, we first build a sentiment emoji lexicon. The lexicon contains both positive and negative emojis expressing the feelings corresponding to different sentiment categories. We collect the emojis as well as their indicated sentiment from "Emojis Sentiment Ranking Lexicon" (Kralj Novak et al., 2015) which is available at `http://kt.ijs.si/data/Emoji_sentiment_ranking/` and Emojipedia[4]. Then, this lexicon is employed as the seed for the Twitter retrieval procedure. The Lexicon is composed of 91 negative emojis and 306 positive emojis.

Instead of collecting tweets by hashtags or query terms we exploit the emojis and their assigned sentiment and condition the tweet language set to Arabic. We extracted 59k of the tweets using the Twitter API in April 2019. The corpus contains multiple dialects from all over the Arab world as it is not geographically constrained. To automatically annotate the tweets either as positive or negative, we use the emojis as a noisy (weak) label. If the tweet is fetched by the positive emojis from the lexicon like ☺ then it is labelled as positive and the tweets fetched by the negative lexicon are labelled as negative.

More specifically, we perform the following cleaning actions:

1. Remove all metadata generated by Twitter API like tweet_id, username, time, location, RT
2. Remove all special characters but not emojis
3. Remove non-Arabic characters
4. Remove links
5. Remove diacritics from the text
6. Remove duplicated tweets

Table 1 shows the statistics of the corpus before and after the pre-processing phase which gives us 36K tweets.

	Positive	Negative	Total	Vocabs	Words
Before	30,607	29,232	59,839	95,538	76,2673
After	18,173	18,695	36,868	95,057	41,8857

Table 1: Statistics of the Twitter sentiment analysis corpus (ATSAD) before and after the pre-processing

5. Corpus Evaluation

The process of building a resource is not limited to data collection, but it must be checked and verified in order to be trustworthy and used as a resource. In this section, we evaluate the Tweets corpus by introducing two well-known methodologies: Intrinsic and extrinsic evaluations.

In intrinsic evaluation, the corpus is directly evaluated in terms of its accuracy and quality. We check whether the rule-based annotation (simply an emojis annotation) can be used to build a reliable corpus and use it effectively in the desired functionality. On the other hand, in extrinsic evaluation, the dataset is going to be assessed with respect to its

[4] https://emojipedia.org/people/emojis

impact on an external task which in our case is the sentiment analysis model (Resnik and Lin, 2010).
To check the quality of the corpus, we have asked two annotators, one an NLP expert, the second an educated native Arabic speaker, to annotate subsets of the corpus. We start with a random sample containing 180 instances (1% of the data) for both positive and negative classes. When the annotation was completed, the two annotators agreed on the 90% of the sample.
In case of disagreement, we choose the expert annotator's choice as the class label. The annotation process is cumulative, in the sense that we pick random samples every time from the corpus and ask the annotators to annotate. For each sample we calculate the number of mismatched labels between the emoji-based annotation and the human annotation, and we also compute the accuracy of the emoji-based annotation by taking the number of right classified instances divided by the total number of the sample. Table 2 shows the number of errors (mismatches) and accuracy for annotation samples in the range from 1% to 10% of the corpus. Figure 1 plots the accuracy results. It is clear that after manually annotating 10% of the whole corpus, the percentage of matches tweets between the human and the emoji-based annotation is 77.2%.
Obtaining 77.2% is not good enough to use it for a task to predict the real sentiment of the tweets even though it is less time-consuming compared to manual annotation. Therefore, later we are going to present a combination method of self training and distant supervision to improve the quality of the dataset.

Sample %	Samples	#errors	Accuracy
1%	360	106	70.5%
2%	720	200	72.2%
3%	1,080	293	72.9%
4%	1,400	370	74.3%
5%	1,800	450	75%
10%	3,608	823	77.2%

Table 2: Human annotation accuracy compared to the emojis based annotation. The first two columns show the percentage and number of the sampled tweets, #_error shows the number of mismatched samples and the Accuracy column calculates the percentage of the matches between both annotations.

Moreover, we check the quality of the corpus with pre-trained sentiment analysis models that have been built and trained on existing datasets. The following datasets are used in our experiments and shown in Table 3:

- 40k dataset (Mohammed and Kora, 2019): as mentioned in the related work section, this is a tweets dataset containing 40,000 instances. It is manually annotated into positive and negative and the tweets are subsequently manually cleaned.

- LABR (Aly and Atiya, 2013): a large SA dataset for Arabic sentiment analysis. The data are extracted from a book review website and contain over 63k book reviews written in MSA with some dialectal phrases. Given that our corpus concerns two-way classification, we only use the binary balanced subsets of LABR. LABR can be considered to be a human annotated corpus, where the users rate books using the stars system (1 to 5).

  Ratings of 4 and 5 stars are considered positive, ratings of 1 and 2 stars negative and 3-star ratings are taken as neutral. In the binary classification case, 3-star ratings are ignored, keeping only the positive and negative labels.

- ASTD (Nabil et al., 2015): an Arabic SA corpus collected from Twitter and focusing on Egyptian Arabic. It consists of approximately 10k tweets which are classified as objective, subjective positive, subjective negative, and subjective mixed. We use only the positive and negative subset.

- Shami-Senti (Qwaider et al., 2019): a Levantine SA corpus. It contains approximately 2.5k posts from social media sites in general topics classified manually as positive, negative and neutral from the four main countries where Levantine is spoken: Palestine, Syria, Lebanon and Jordan.

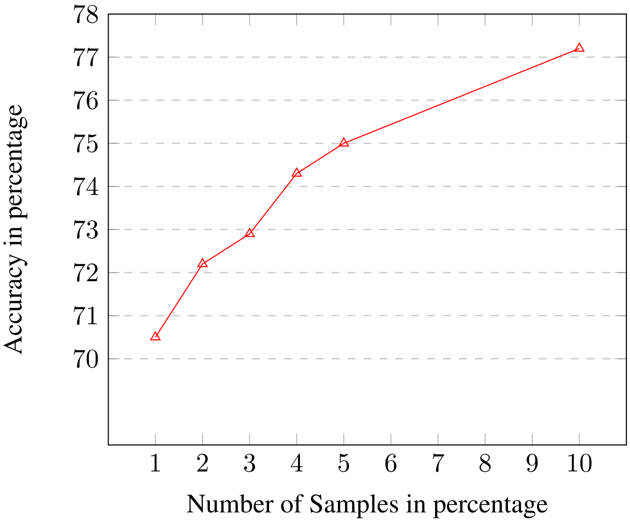


Figure 1: Accuracy of dataset comparing to human annotation

Corpus	NEG	POS
40k tweets	20,002	19,998
LABR 2 Balanced	6,578	6,580
ASTD	1,496	665
Shami-Senti	935	1,064

Table 3: The number of instances per category in the corpora used in our experiments

We build a model on each corpus and apply the resulting model to our Twitter corpus. The model uses a combination of (1-3) word grams and a LinearSVC classifier. Table 4 shows the accuracy of the models built (trained and tested) on the original datasets, while the ASTAD column shows the accuracy of the trained model when we use it to predict the class on our Twitter dataset. It is clear that none of the models works for this dataset and the accuracy does not exceed 60%. This is an expected result, given that the data are from a very different domain, i.e. book reviews. Even though both ASTD and the ATSAD share the same domain, the ASTD only contains Egyptian dialects. In the case of the Shami corpus, it only contains Levantine dialects with a limited number of examples (2k). The 40k tweets model and ATSAD also share the same domain (tweets) but the manual hard prepossessing and cleaning of the data make it hard to predict real tweets as people post it, also the 40k corpus only has Egyptian dialect.

	Same corpus	ATSAD
40k tweets	79%	60%
LABR	82%	54%
ASTD	81%	59%
SHAMI-SENTI	84%	59%

Table 4: Accuracy of models trained on different SA corpora; the same corpus column indicates the accuracy of the model when the train dataset and the test dataset are both from the same corpus, the last column for the accuracy when we test the models on the ATSAD

Summing up, it is clear from the previous discussion that the ATSAD is a challenge for the models trained on the available datasets that are standardised and regularised. Therefore we have to create an ML model that would be successful on this ATSAD. To achieve a good accuracy on the model, then the dataset should be improved in term of the data quality and annotation quality.

6. Self training on Distant supervision Corpus

Creating a good resource requires the collection of a big amount of data that are preprocessed and annotated. The annotation is usually done by hiring annotators and specifying annotation rules they have to follow to produce a reasonable annotation agreement. This process is time and money consuming. There is another approach to build a large enough dataset more quickly. The process is called Distant supervision or weak supervision (Yao et al., 2010). Distant supervision involves heuristically matching the contents of a database to the corresponding text (Hoffmann et al., 2011). In our case, we use the emojis in the tweets to work as weak labels with which we can annotate the 36K tweets automatically. Although this is sometimes not producing high-quality dataset, it works in some tasks.

We annotate the 36k tweets by distant supervision and then extract 4k tweets (10% of the total dataset). We ask the two annotators to label them manually. We compute the number of agreed annotation between the human annotation and the emojis annotation we have an agreement of 77.2%.

To use the human annotation dataset as a gold standard we extract other 4K tweets and also manually annotate them, upgrading the final manually annotated dataset to 8k tweets of which 3705 are classified as positive, 3911 negative and 384 instances are mixed. We exclude the mixed class from our experiments.

We build a baseline with TF-IDF unigram word model and a Linear-SVC classifier. Moreover, we build another complex model -from some previous work - by combining word n-grams (1-5), character n-grams (2-5) with and without word boundary consideration (Qwaider et al., 2019). The models are built for sentiment analysis and the problem is recognised as two-way classification, so every tweet is classified either as positive or negative. Table 5 shows the number of tweets per class for the human annotation dataset and the remaining tweets in the emojis dataset which were weakly annotated by the distant supervision.

	Human annotated	Emojis annotated
	Label Distribution	
#Positive	3,705	14,468
#Negative	3,911	14,784
	Train/Test Distribution	
#Train_set	6,092	23,401
#Test_set	1,524	5,851
#Total_set	7,616	29,252

Table 5: Statistics of the human annotation subset and the emojis distant supervision subset after subtract the human dataset

We apply both the baseline and the complex model on the manually annotated dataset and we get an accuracy of %71 and %79 respectively. We refer to this experiment as (Manual experiment). To check again the quality of the emojis based dataset we applied the previous model trained on the human labels on the emojis dataset of 29k tweets to predict the label. After testing the two models, the resulted accuracy is %63 and %76 for both the baseline and the complex model respectively (Mixed experiment). The mixed experiment is to some extent similar to the agreement between the manual annotation and the emojis annotation experiment we have done first and got an accuracy of 76% using 4k subset.

To improve the quality of the automatic annotation and therefore the proposed tweets corpus, we will exploit the manual annotation dataset to enhance the entire dataset. Therefore, a self-training approach is to be employed on the data to improve the classification and increase the accuracy of the annotation. Self-training is a commonly used method for semi-supervised learning (Yarowsky, 1995; Abney, 2002). The idea of Self-training is to train a classifier with a small amount of labelled data and incrementally retrain the classifier by adding the most confidently labelled instances that were previously unlabelled as a new data. This process continues until most of the unlabelled data becomes labelled (Gao et al., 2014). We can implement a self-training technique with little modification of the existing configuration: our dataset is not completely unlabelled but has weak emoji-based annotations. From the mixed model experiments, rather than extracting the instances predicted with the highest confidence, we extract instances where the

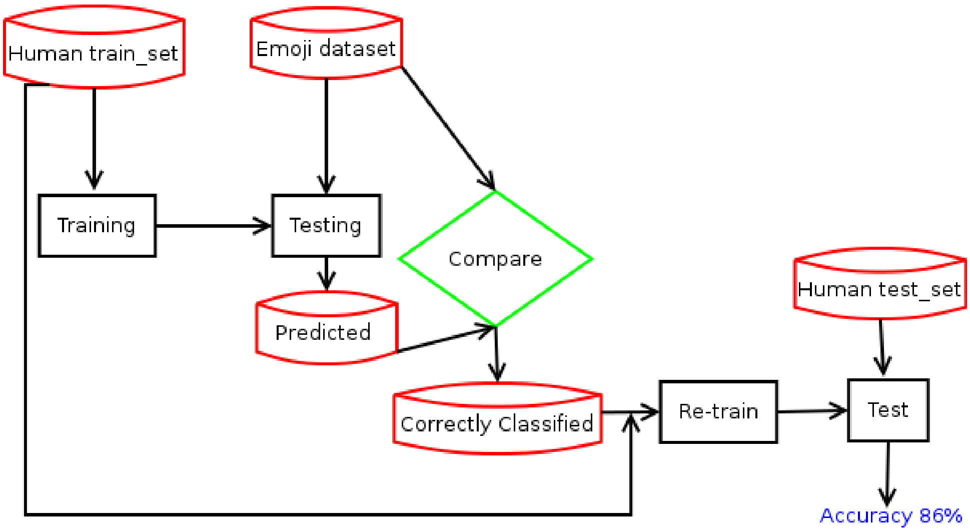


Figure 2: Self training (double-check) approach applied on the TSAD

model prediction label matches the emojis label. This is the case for 22,542 out of 29,252 tweets in the dataset. We add these tweets to the training set which consists of the human annotated dataset (6,092). Thus, to re-train the classifier we have a total of 22,542 + 6,092 = 28,634 tweets. We call this experiment (double check) where we combine the self training with distant supervision. The 28K tweets are now a dataset with strong supervised labelling where the small amount of human annotation dataset and distant supervising from emojis helps to annotate more data. We re-build both the baseline and the complex models and retrain them on the dataset we produced from the double check experiment (28k tweets), then apply the model to the test set from the human annotation dataset (1,524 tweets).We use the same dataset across all the experiments in order to allow for the comparison. The baseline and the complex model accuracy increases to 77% and 86% respectively. Figure 2 shows the diagram for the self-training approach.
To evaluate our self-training experiment and our method to extract only those instances where the model prediction matches the emojis annotation, we conduct a small experiment of self-training called (Non-check) where we:

1. Use the model from the (mixed experiment) to predict the label for the automatically labelled dataset (29k tweets).

2. Retrain the model with the human annotated training dataset in addition to the predicted labelled dataset (from the previous model). Thus, this re-train dataset consists of 6,092 + 29,252 = 35,341 tweets.

3. Use the manually annotated test set (1524 tweets) and use the model to predict the sentiment.

4. The accuracy of the baseline is 70% and 81% for the complex model.

Consequently, it is clear that (i) using the emojis as a noisy label, (ii) matching with the human annotation and (iii) apply the self training technique to annotate the dataset leads to an improvement of the data. Table 6 shows the performance of the models on different datasets. These are represented as plots in Figure 3.

Experiment	#Train	#test	Baseline	Complex
Manual	6,092	1,524	71%	79%
Mixed	6,092	29,252	63%	76%
double-check	28,634	1,524	**77%**	**86%**
Non-check	35,341	1,524	70%	81%

Table 6: The performance of the baseline and complex models on different datasets.

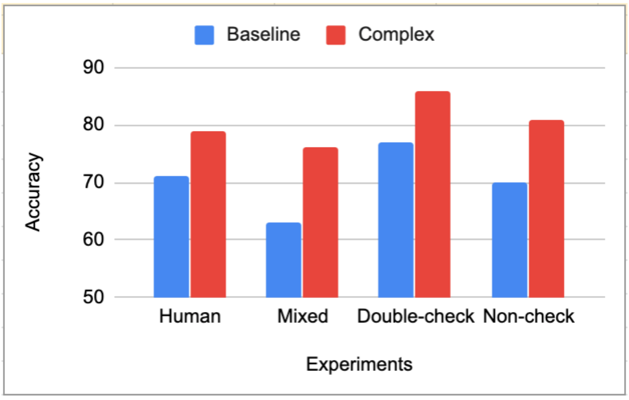
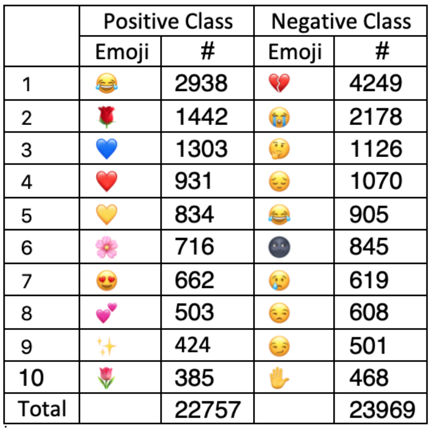


Figure 3: plotting the accuracy for all the experiments for both the baseline and complicated models

When we were done with the experiments, we extracted all the emojis and examined the emoji frequencies per category. We found some emojis are shared between the positive and the negative class, such as the smiley face with tears. We also discover that people used the black smiley face to indicate the negative feeling more often than the positive. These emojis are considered tricky emojis and they decrease the quality of the annotation. We modified our conditions by removing all the misleading emojis to collect more accurate data. Up to now we have collected over 200k tweets. Table 7 views the number of occurrence for the most 10 frequent emojis per sentiment category.



	Positive Class		Negative Class	
	Emoji	#	Emoji	#
1	😂	2938	💔	4249
2	🌹	1442	😭	2178
3	💙	1303	🤔	1126
4	❤️	931	😔	1070
5	💛	834	😂	905
6	🌸	716	🌚	845
7	😍	662	😢	619
8	💕	503	😒	608
9	✨	424	😏	501
10	🌷	385	✋	468
Total		22757		23969

Table 7: Number of occurrence for the most 10 frequent emojis per category, the last row show the total number of the whole emojis in the dataset per category

7. Future work

Based on our emojis analysis and the subsequent modification of the data collection and annotation conditions, we are planning to further increase the size of the dataset and use it for different tasks like building custom sentiment word embeddings and to fine-tune deep learning networks.

8. Conclusion

To extend the limited Dialectal Arabic resources, we collected an Arabic Tweets Sentiment Analysis Dataset (ATSAD). The corpus has been collected from Twitter during April 2019 and employs emojis as seeds for extraction of candidate instances. After the pre-processing, we apply distant supervision using emojis as weak labels to annotate the entire dataset. In addition, we commissioned two annotators to manually annotate a subset of 8k tweets. We evaluate the corpus by comparing the emoji-based annotation with the human annotation and we get an observed agreement of 77.2%. We built a sentiment analysis machine learning model with the unigram features as a baseline and another complex model that utilises word grams and character grams. We exploit the human annotation dataset to help us improve the annotation of the automatically labelled dataset by self-training approaches. Over several experiments we achieve an accuracy of 86%.

Using the distant supervision approaches for automatically data annotation process can saves us a lot of effort, time and money. Distant supervision is a very valuable method to annotate large number of instances automatically, in our case based on emojis to denote the category. The self training approach can be used together with a small number of manually annotated instances to improve the quality of the automatically labelled dataset.

Acknowledgements

Kathrein Abu Kwaik, Stergios Chatzikyriakidis and Simon Dobnik are supported by grant 2014-39 from the Swedish Research Council, which funds the Centre for Linguistic Theory and Studies in Probability (CLASP) in the Department of Philosophy, Linguistics, and Theory of Science at the University of Gothenburg. Richard Johansson was funded by the Swedish Research Council under grant 2013–4944.

9. Bibliographical References

Abdul-Mageed, M., Diab, M., and Kübler, S. (2014). Samar: Subjectivity and sentiment analysis for Arabic social media. *Computer Speech & Language*, 28(1):20–37.

Abney, S. (2002). Bootstrapping. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 360–367.

Al Sallab, A., Hajj, H., Badaro, G., Baly, R., El Hajj, W., and Shaban, K. B. (2015). Deep learning models for sentiment analysis in Arabic. In *Proceedings of the second workshop on Arabic natural language processing*, pages 9–17.

Al Shboul, B., Al-Ayyoub, M., and Jararweh, Y. (2015). Multi-way sentiment classification of Arabic reviews. In *2015 6th International Conference on Information and Communication Systems (ICICS)*, pages 206–211. IEEE.

Al-Twairesh, N., Al-Khalifa, H., and Al-Salman, A. (2016). AraSenTi: Large-Scale Twitter-Specific Arabic Sentiment Lexicons. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 697–705, Berlin, Germany, August. Association for Computational Linguistics.

Al-Twairesh, N., Al-Khalifa, H., Al-Salman, A., and Al-Ohali, Y. (2017). Arasenti-tweet: A corpus for Arabic sentiment analysis of saudi tweets. *Procedia Computer Science*, 117:63–72.

Alayba, A. M., Palade, V., England, M., and Iqbal, R. (2018). A combined CNN and LSTM model for Arabic sentiment analysis. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pages 179–191. Springer.

Alwakid, G., Osman, T., and Hughes-Roberts, T. (2017). Challenges in sentiment analysis for Arabic social networks. *Procedia Computer Science*, 117:89–100.

Aly, M. and Atiya, A. (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)*, volume 2, pages 494–498.

Atoum, J. O. and Nouman, M. (2019). Sentiment Analysis of Arabic Jordanian Dialect Tweets. *(IJACSA) International Journal of Advanced Computer Science and Applications*, 10:256–262.

Azmi, A. M. and Alzanin, S. M. (2014). Aara'–a system for mining the polarity of Saudi public opinion through e-newspaper comments. *Journal of Information Science*, 40(3):398–410.

Bakliwal, A., Arora, P., Madhappan, S., Kapre, N., Singh, M., and Varma, V. (2012). Mining sentiments from tweets. In *Proceedings of the 3rd Workshop in Computational Approaches to Subjectivity and Sentiment Analysis*, pages 11–18.

Bravo-Marquez, F., Mendoza, M., and Poblete, B. (2013). Combining strengths, emotions and polarities for boosting twitter sentiment analysis. In *Proceedings of the Second International Workshop on Issues of Sentiment Discovery and Opinion Mining*, page 2. ACM.

Dahou, A., Xiong, S., Zhou, J., Haddoud, M. H., and Duan, P. (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.

El-Beltagy, S. R., Khalil, T., Halaby, A., and Hammad, M. (2016). Combining lexical features and a supervised learning approach for Arabic sentiment analysis. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 307–319. Springer.

Elnagar, A., Khalifa, Y. S., and Einea, A. (2018). Hotel Arabic-reviews dataset construction for sentiment analysis applications. In *Intelligent Natural Language Processing: Trends and Applications*, pages 35–52. Springer.

Gao, W., Li, S., Xue, Y., Wang, M., and Zhou, G. (2014). Semi-supervised sentiment classification with self-training on feature subspaces. In *Workshop on Chinese Lexical Semantics*, pages 231–239. Springer.

Giachanou, A. and Crestani, F. (2016). Like It or Not: A Survey of Twitter Sentiment Analysis Methods. *ACM Comput. Surv.*, 49(2):28:1–28:41, June.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, 1(12):2009.

Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., and Weld, D. S. (2011). Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 541–550. Association for Computational Linguistics.

Kralj Novak, P., Smailović, J., Sluban, B., and Mozetič, I. (2015). Sentiment of emojis. *PLoS ONE*, 10(12):e0144296.

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

Mohammed, A. and Kora, R. (2019). Deep learning approaches for Arabic sentiment analysis. *Social Network Analysis and Mining*, 9(1):52.

Nabil, M., Aly, M., and Atiya, A. (2015). ASTD: Arabic Sentiment Tweets Dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2515–2519.

Obaidat, I., Mohawesh, R., Al-Ayyoub, M., Mohammad, A.-S., and Jararweh, Y. (2015). Enhancing the determination of aspect categories and their polarities in Arabic reviews using lexicon-based approaches. In *2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, pages 1–6. IEEE.

Qwaider, C., Chatzikyriakidis, S., and Dobnik, S. (2019). Can Modern Standard Arabic Approaches be used for Arabic Dialects? Sentiment Analysis as a Case Study. In *Proceedings of the 3rd Workshop on Arabic Corpus Linguistics*, pages 40–50.

Refaee, E. and Rieser, V. (2014). Evaluating distant supervision for subjectivity and sentiment analysis on Arabic twitter feeds. In *Proceedings of the EMNLP 2014 workshop on Arabic natural language processing (ANLP)*, pages 174–179.

Resnik, P. and Lin, J. (2010). 11. evaluation of NLP systems. *The handbook of computational linguistics and natural language processing*, 57.

Rushdi-Saleh, M., Martín-Valdivia, M. T., Ureña-López, L. A., and Perea-Ortega, J. M. (2011). OCA: Opinion Corpus for Arabic. *Journal of the American Society for Information Science and Technology*, 62(10):2045–2054.

Saif, H., He, Y., and Alani, H. (2012). Alleviating data sparsity for twitter sentiment analysis. CEUR Workshop Proceedings (CEUR-WS. org).

Speriosu, M., Sudan, N., Upadhyay, S., and Baldridge, J. (2011). Twitter polarity classification with label propagation over lexical links and the follower graph. In *Proceedings of the First workshop on Unsupervised Learning in NLP*, pages 53–63. Association for Computational Linguistics.

Yao, L., Riedel, S., and McCallum, A. (2010). Collective cross-document relation extraction without labelled data. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1013–1023. Association for Computational Linguistics.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *33rd annual meeting of the association for computational linguistics*, pages 189–196.