ISSN 2007-9737

Tunisian Dialect Sentiment Analysis: A Natural Language Processing-based Approach

Hala Mulki[1], Hatem Haddad[2], Chedi Bechikh Ali[2], Ismail Babaoğlu[1]

[1] Selcuk University, Department of Computer Engineering,
Turkey

[2] Université Libre de Bruxelles, Department of Computer & Decision Engineering (CoDE),
Belgium

[3] Carthage University, LISI Laboratory, INSAT,
Tunisia

halamulki@selcuk.edu.tr, hatem.haddad@ulb.ac.be, chedi.bechikh@gmail.com, ibabaoglu@selcuk.edu.tr

Abstract. Social media platforms have been witnessing a significant increase in posts written in the Tunisian dialect since the uprising in Tunisia at the end of 2010. Most of the posted tweets or comments reflect the impressions of the Tunisian public towards social, economical and political major events. These opinions have been tracked, analyzed and evaluated through sentiment analysis systems. In the current study, we investigate the impact of several preprocessing techniques on sentiment analysis using two sentiment classification models: Supervised and lexicon-based. These models were trained on three Tunisian datasets of different sizes and multiple domains. Our results emphasize the positive impact of preprocessing phase on the evaluation measures of both sentiment classifiers as the baseline was significantly outperformed when stemming, emoji recognition and negation detection tasks were applied. Moreover, integrating named entities with these tasks enhanced the lexicon-based classification performance in all datasets and that of the supervised model in medium and small sized datasets.

Keywords. Tunisian sentiment analysis, text preprocessing, named entities.

1 Introduction

Social media users tend to use informal language to express their opinions. Arabic informal language combines a variety of dialects differ from each other such that same words or expressions may have drastically different sentiments. During and after the Tunisian revolution, tracking the public reactions and impressions against different events has been conducted through sentiment analysis systems [2].

Previous work on Tunisian Sentiment Analysis (SA) has mostly processed the textual data using initial cleaning and normalization procedures [18, 11, 8].

Although these models have achieved quite satisfying results, improving the sentiment classification by the application of further preprocessing tasks remains an interesting field of research. Furthermore, we believe that exploiting sentiment indicative words derived from the corpus itself such as Named Entities (NEs) and including them as a preprocessing step can contribute in inferring the sentiment in the subsequent sentiment classification step. Indeed, the shared online opinions are rich of all types of Named Entities (persons, locations or organizations) towards which sentiment is expressed [19].

We assume that Named Entity Recognition (NER) task can be exploited in sentiment analysis if the extracted NEs were sentimentally classified

doi: 10.13053/CyS-22-4-3009

according to the local context in which they are mentioned.

To the best of our knowledge, NEs have not been included before in Tunisian SA systems. In this paper, we aim at improving the performance of Tunisian SA by the single or combined application of the following Natural Language Processing (NLP) techniques: stopwords removal, stemming, negation detection and common emoji recognition. Moreover, we introduce named entities tagging as a preprocessing task and investigate its impact on the sentiment classification performance when it is combined with other preprocessing tasks.

To do that, three different-sized Tunisian datasets containing positive/negative tweets and comments from multiple domains provided by [18, 11, 8] were used. The data has been subjected to several combinations of preprocessing techniques with NEs tagging integrated. Then, Tw-StAR SA system [13] that includes two model variants: supervised learning-based and lexicon-based has been employed to perform a sentence-level sentiment analysis of the tackled data. Finally, we compared our results with those of [18, 11, 8], who applied sentence-level SA on the same datasets used here.

2 Tunisian Sentiment Analysis

Tunisian sentiment analysis can be conducted using machine learning (ML) approaches such as supervised methods or lexicon-based approaches.

– Supervised learning-based method: This method requires a labeled corpus to train the classifier to predict the text polarity [14]. The learning process is carried out by inferring that a combination of a sentence's specific features yields a specific polarity class: positive, negative. The features used in this strategy are bag-of-N-grams features. Having the features extracted, sentiment classification is then performed using several supervised classification algorithms such as Support Vector Machines (SVM), Naive Bayes (NB), Logistic Regression (LR), K-Nearest Neighbors (KNN), etc.

– Lexicon-based method: In the lexicon-based model, neither labeled data nor a training step are required to design the sentiment classifier. The sentiment embedded in a sentence or a document is determined with the assistance of manually-built, predefined or translated sentiment lexicons. A sentiment lexicon contains subjective words along with their polarities (positive or negative) and polarity scores; the so called weights [14]. Thus, the polarity of a word or a sentence can be decided due to its sentimental score derived from the lexicon.

Modern Standard Arabic (MSA) is the formal Arabic language. It has a complex morphology to handle, where words are of highly inflectional and derivational nature. Informal or dialectal Arabic combines a wide variety of dialects each of which has its own grammar, writing styles and slang terms. This makes dialectal Arabic SA task more challenging and requires the customization of the existing NLP tools to be able to address the properties of each dialect [10]. Tunisian dialect or “Darija” has been recently tackled in some SA research such that multiple datasets and SA systems were provided.

To enrich the dialectal Arabic SA resources with Tunisian corpora, [18] collected a dataset composed of 5,514 MSA/Tunisian tweets related to the election context. The manually annotated tweets were classified using six different classifiers trained with multiple schemes of n-grams features. Two types of classification were performed: binary classification for positive/negative tweets and multi-class classification with neutral tweets considered. The best performance for binary classification was achieved by SVM algorithm with an accuracy of 71.09% and F1-score of 63%.

The authors in [11] presented the first attempt to employ document embeddings as features in a Tunisian SA model. Their model was evaluated using a combination of publicly available MSA/multi-dialectal datasets: OCA [16], LABR [3] and a manually annotated Tunisian Sentiment Analysis Corpus (TSAC) obtained from Facebook comments about popular TV shows.

Doc2vec algorithm was applied to generate document vectors of each comment. The produced vectors were used to train SVM, Bernoulli NB (BNB) and Multilayer Perceptron (MLP) classifiers. The best results were scored by MLP classifier when TSAC corpus was solely used as a training set where it achieved an accuracy equals to 78% and an F1-score of 78%.

In [8], a lexicon-based SA system to classify the opinions embedded in Tunisian customers' reviews was proposed. To support identifying the Tunisian dialect, the author developed a Tunisian Arabic morphological analyzer along with a transliteration machine to handle the arabizi form of the Tunisian dialect. In addition, a Tunisian version of the SentiWordNet lexicon was used. The preprocessing phase employed the developed morphological analyzer to produce various morphological features. The model was applied on positive, negative and neutral tweets which belong to multiple domains. The classification accuracy of the used Tunisian lexicon was 72.1% considering only the positive/negative tweets.

3 NLP for Arabic Sentiment Analysis

Considering the research dedicated to Arabic dialects, few works focused on the Tunisian dialect. Dialectal Arabic is usually manipulated using MSA based NLP methods.

In [5], the authors have investigated the impact of several feature representation models along with NLP preprocessing techniques on SA of positive/negative MSA and dialectal reviews. The effect of stemming and light stemming combined with stopwords removal was evaluated using three ML classifiers: NB, SVM, and KNN trained using RapidMiner. Two datasets were used; the first one contained 300 dialectal political reviews that harvested from Aljazeera website. The second dataset, however, combined an MSA content of 500 movie reviews [16]. The experimental study concluded that combining stopwords removal with both stemming and light stemming could not improve the performance as the built-in stemmers provided by RapidMiner tend to have high error rates.

With stemming applied, KNN classifier applied on OCA dataset achieved an accuracy of 82.2% compared to 89.6% scored by word n-grams features. Stemming techniques have been also investigated in [4] as root-stemming and light stemming were applied together with tokens, n-grams characters and feature selection methods to improve the SA of MSA/Jordanian tweets. The impact of the preprocessing on SA has been studied through the application of a root-stemmer and a light-stemmer on a dataset of 2000 positive/negative tweets. SVM, NB and KNN classifiers were used for training. It has been referred that when SVM feature selection method was applied on stemmed feature words, light-stemmer was better than root-stemmer with an accuracy of 87.65% compared to 86.85% achieved for the root-stemmed data.

More recently, [6] presented a hybrid SA model in which data was subjected to light stemming, mention/emoji detection and tagging before classifying the sentiment. In addition to the previous preprocessing tasks, character normalization, elongation removal, URLs and punctuation detection tasks were also applied. N-gram features and lexical features were employed to train a Complement Naïve Bayes classifier. Using this model, the authors classified the sentiment of positive,negative and neutral multi-dialectal Arabic tweets within the context of SemEval-2017 workshop [15]. The official ranking results indicted the outperformance of this model over the other competing systems as it ranked first with an accuracy of 58.1%.

4 Proposed Sentiment Analysis Framework

In this study, we aim to define which one or a combination of stemming, light-stemming, stopwords removal, common emotions and negation detection tasks can efficiently improve the Tunisian SA performance. Consequently, we can decide with which preprocessing task/tasks named entities tagging should be combined such that the sentiment classification performance could be better enhanced.

Sentiment analysis of Tunisian data has been tackled using Tw-StAR framework which includes two models: supervised and lexicon-based. NEs were extracted using the system provided by [7] then classified and tagged as positive or negative to be used in an additional preprocessing task in both models. N-grams schemes were adopted as features to train the supervised classification algorithms in the supervised model while these features were looked up in the sentiment lexicon adopted by the lexicon-based model.

4.1 Preprocessing

The pipeline used for the preprocessing can be briefed via the following steps:

– Initial preprocessing: For all datasets, a common initial preprocessing step that includes removing the non-sentimental content such as URLs, usernames, dates, digits, hashtags symbols, and punctuation was performed.

– Stemming (Stem): A stemmer is used to strip a word's suffix and prefix such that the variation of word morphology can be handled. To study the effect of stemming algorithms on Tunisian Sentiment Analysis, we investigated stemming and light stemming.

  With the absence of a Tunisian stemmer, Farasa [1] stemmer was used because it yields lower segmentation errors than existing Arabic stemmers. The main idea of FARASA is to use SVMrank to rank possible ways to segment words to prefixes, stems, and suffixes. For example, the word "wktAbnA" "و كتابنا" meaning: "and our book" is composed of three clitics "w+ktAb+nA", namely the conjunction article "w" as prefix, the stem "ktAb", and possessive pronoun "nA" as suffix. The underlying idea is to eliminate segmented affixes. Unlike stemming, light stemming removes common affixes from words without reducing them to their stems or roots. In our study, we used the light stemmer provided by [9].

– Stopwords (Stop): To remove common Arabic stopwords, a list of 1,661 MSA stopwords provided by the NLP group at the National Center for Computing Technology and Applied Mathematics in King Abdulaziz City for Science and Technology (KACST)[1] was used.

– Common Emotions Detection (Emoji): We have identified two types of the most common emoji. The first type refers to positive emoji such as smiling face, grinning face, kissing face, etc. The second type represents the negative emoji such as unamused face, pensive face, worried face, etc. Positive emoji were replaced with the label "PositiveEmoji" while the tag "NegtaiveEmoji" was used to replace negative emoji.

– Negation Detection and Tagging (Neg): Negation was inferred from the negative words: ”لا”, ”لم”, ”لن”, ”لستُ”, ”ليس”, ”دون”,”لسن”, ”ليسوا”, ”بدون”, ”بلا”, ”أبداً”, ”بغير”, ”غير”, ”لستم”, ”لستن”. Five Tunisian-specific negative indicators were also used: ”ماكش” (Makech), ”مانيش” (Manich), ”ماكمش” (Makomch), ”مفماش” (Mafamech) and ”ماهمش” (Mahomch). Four indicators appear before verbs: ”لا”, ”لم”, ”لن”, ”ما”. The negatives particles ”ليس” (layssa) appears before a noun phrase or a verb phrase: ”لستُ” (lasstou), ”لسن” (lassna), ”ليسوا” (layssou), ”لستم” (lasstom), ”لستن” (lasstona) are variation of ”ليس” which mean not. ”بلا” came before a name, for example ”بلا لعب”, which mean without (بلا) joking (لعب). We have used the tag "NegWord" to replace each of the previous words.

– Feature Extraction: Having the data preprocessed, it was subjected to tokenization to generate N-grams features. Three N-grams schemes including unigrams, bigrams and trigrams were adopted in the supervised model as they can capture information about the local word order and save the training time consumed by supervised methods. For a certain N-grams scheme, a tweet's feature vector is constructed via examining the presence or absence of this scheme among the review's

[1] https://github.com/abahanshal/arabic-stop-words-list1

tokens. Consequently, the feature vector's values are identified as True (presence) or False (absence). Term frequency property was employed to reduce the feature size according to predefined frequency thresholds. Regarding the lexicon-based model, unigrams and a combination of unigrams and bigrams were used in order to cover single and compound phrases of the used lexicon.

4.2 Named Entities Recognition

Named entities were processed using the NER system provided by [7]. It is based on deep neural networks as it combines a Bi-directional Long Short Term Memory with a Conditional Random Fields. In addition, the system uses character-level representation of words together with the pretrained word embeddings to initiate the word representation vectors to deal with the out-of-vocabulary issues.

The produced named entities were then classified into positive or negative in order to be tagged in the preprocessing step. For this purpose, we have developed a polarity assignment algorithm through which the polarity of an NE within a corpus is determined based on its local contextual information as following:

– NEs extracted from the training datasets are compared against the sentence's tokens included in the training set.

– When a match between a specific NE and a sentence is found, an aggregated score is assigned to this NE due to the polarity of that sentence such that 1 is added if the sentence's polarity is positive while 1 is subtracted if the sentence is of negative polarity.

– Thus, the polarity of a certain NE is determined by the sign of its accumulated resulted score where positive and negative signed scores define positive and negative NEs respectively.

– As for NEs of zero-valued scores, they have been eliminated as they have been mentioned equally in positive and negative sentences.

The positive/negative NEs resulted from the previous step were looked up in datasets and tagged as “posNE” or “negNE”.

4.3 Sentiment Classification

The supervised SA model is trained to predict the proper polarity class relevant to specific input features. Training has been performed first with NB algorithm from scikit-learn then using linear SVM from LIBSVM as SVM can handle high-dimensional feature vectors effectively.

To determine the polarity of a tweet via the lexicon-based model, Straight Forward Sum (SFS) method, with the constant weight strategy as negative and positive words have the weight of -1 and 1, respectively, was used as in [12]. The polarity of a given sentence is thus calculated by accumulating the weights of negative and positive terms contained in it. Consequently, the sentence polarity is determined by the sign of the resulted sum. A manually-built Tunisian sentiment lexicon of 5,382 entries was used with non-stemmed data while for input data being stemmed or light stemmed, this lexicon was extended to include the stemmed/light-stemmed variations of its words and phrases such that the lexicon size was increased into 14,345 single and compound entries.

5 Experimental Study

In the presented tables, performances obtained for several single/combinations of preprocessing tasks are compared against the baselines that represent the performances achieved by the systems of [18], [8] and [11] respectively. Precision, recall, F1-score and accuracy are referred to as (P.), (R.), (F1.) and (Acc.) respectively.

5.1 Datasets

Three publicly available datasets with a content harvested from Tunisian or mixed Tunisian-MSA social media platforms have been used:

- Tunisian Election Corpus (TEC): Refers to a set of 5,521 tweets collected by [18] during the Tunisian elections period in October 2014. It combines MSA and Tunisian dialect where Tunisian tweets form the majority of the data. After reducing neutral tweets, a dataset of 3,043 tweets was used.

- Tunisian Sentiment Analysis Corpus (TSAC): A dataset of 9,976 Facebook comments provided by [11]. These comments represent the reactions of the audience towards popular Tunisian TV shows, they were annotated manually for positive and negative polarity. In this study, we filtered the Arabizi instances out of this dataset such that 7,366 comments were used.

- Tunisian Arabic Corpus (TAC): A dataset composed of 800 tweets which cover multiple topics such as media, telecom and politics. This dataset have been collected by [8] and annotated for positive, negative and neutral polarity. We have only handled the positive and negative instances such that 746 tweets were adopted.

Negative words statistics and the detailed statistics of the polarity distribution across these sets are reviewed in Table 1.

5.2 Results and Discussion

The preprocessing techniques listed in Section 3 have been examined one by one then different combinations of them have been applied. This enabled defining the preprocessing technique/combination for which the SA performance is better improved and hence specifying the preprocessing technique/combination in which NEs tagging could be integrated.

In the supervised model, three variants of experiments were conducted. The first one involved using all N-grams features: unigrams (uni), bigrams (bi), trigrams (tri) and combinations of them (uni+bi, uni+bi+tri), while the second and third experiments used a reduced number of the same features resulted from applying term frequency property with two threshold values equal to 2 and 3 respectively. Table 2, Table 4 and Table 6 list the best performances achieved by either NB or SVM in the supervised model.

The results in Table 6 clearly suggest that SVM always performs better than NB for large-sized datasets such as TSAC. This could be explained by the ability of SVM to handle the sparsity and high-dimensionality of the training feature vectors. However, Table 2 and Table 4 show that the sentiment in medium and small-sized datasets (TEC, TAC) is better classified by NB.

As for the lexicon-based model, each tweet/comment was tokenized into unigrams (uni) then into a combination of unigrams and bigrams (uni+bi) to be looked up later in the manually-built Tunisian lexicon. The polarity score of the input sentence was then calculated using SFS method. Table 3, Table 5 and Table 7 list the best performances achieved by the lexicon-based model.

In these tables, the results obtained for several single/combinations of preprocessing tasks were compared against the baselines that represent the performances achieved by the systems of [18], [8] and [11], respectively; Where Precision, Recall, F1-measure and Accuracy are referred to as (P.), (R.), (F1.) and (ACC.) respectively.

It has been noted that stemming using Farasa improved the supervised sentiment classification performance in TEC, TAC datasets (Table 2, Table 4) where it achieved the second best F1-score (85.9%) in TAC outperforming the baseline by 18.6%. Although Farasa was trained with MSA corpora, it succeeded in identifying the affixes to be cut in Tunisian words because of the lexical overlap between MSA and Arabic dialects in general [17]. In order to retain the variety of words having same root and different meanings, we have also used light stemming. Nevertheless, it failed to increase the evaluation measures in all datasets even when it was combined with other preprocessing techniques.

The impact of stopwords removal on sentiment analysis was revealed when stopwords elimination was combined with stemming. As it can be seen in Table 6, with SVM classifier applied on TSAC, stopwords removal led to a better stemming and thus to a second best F1-score equals to 93.8%

Table 1: Negative words statistics and Training/Test datsets polarity distribution across the used datasets

Dataset	**Negation**		**Training**		**Test**		**Total**
	Training	**Test**	**positive**	**negative**	**positive**	**negative**	
TAC	69	28	306	290	76	74	746
TEC	308	28	968	1466	276	333	3043
TSAC	555	176	2782	3451	672	890	7366

Table 2: The performances of supervised Tw-StAR in TEC dataset with preprocessing

Preprocessing	**Features**	**Model**	**P.(%)**	**R.(%)**	**F1.(%)**	**Acc.(%)**
	uni+bi	SVM [18]	67	71	63	71.1
Stop	uni	SVM	72	70.5	70.6	71.6
Stem	uni	NB	75.3	73.4	73.6	74.5
Neg	uni+bi	SVM	**75.7**	71.7	71.7	73.4
Stem + Stop	uni	NB	**75.7**	73.3	73.4	74.5
Stem + NEs	uni	NB	**75.7**	**74**	**74.2**	**75**

Table 3: The performances of lexicon-based Tw-StAR in TEC dataset with preprocessing

Preprocessing	**Features**	**Model**	**P.(%)**	**R.(%)**	**F1.(%)**	**Acc.(%)**
	uni+bi	SVM [18]	67	71	63	71.1
Stop	uni+bi	Lex	66.6	61.5	59.8	64
Stem	uni+bi	Lex	67.2	64.9	64.5	66.5
Neg	uni+bi	Lex	68.1	62.3	60.5	64.9
Stem + Stop	uni+bi	Lex	67.1	65.7	65.7	67
Stem + NEs	uni+bi	Lex	**68.1**	**68.2**	**67.8**	67.8

Table 4: The performances of supervised Tw-StAR in TAC dataset with preprocessing

Preprocessing	**Features**	**Model**	**P.(%)**	**R.(%)**	**F1.(%)**	**Acc.(%)**
	morphological	Lex [8]	63	72.9	67.3	72.1
Stop	uni	NB	82.9	79.8	79.5	80
Stem	uni	SVM	86.3	85.9	85.9	86
Neg	uni+bi	SVM	86.6	85.9	85.9	86
Stem + Stop	uni+bi	NB	83.9	82.5	82.5	82.7
Neg + NEs	uni+bi	SVM	**87.4**	**86.6**	**86.6**	**86.7**

while in Table 2, the accuracy in TEC dataset was increased from 71.6% scored by stopwords solely to 74.5% achieved by stemming combined with stopwords removal.

Table 5: The performances of lexicon-based Tw-StAR in TAC dataset with preprocessing

Preprocessing	**Features**	**Model**	**P.(%)**	**R.(%)**	**F1.(%)**	**Acc.(%)**
	morphological	Lex [8]	63	72.9	67.3	72.1
Stop	uni+bi	Lex	65	64.8	64.5	64.7
Stem	uni+bi	Lex	65.3	65.3	65.3	65.3
Neg	uni+bi	Lex	69.1	68.8	68.6	68.7
Stem + Stop	uni+bi	Lex	62.4	62.1	61.8	62
Stem + NEs	uni+bi	Lex	**74**	**74**	**74**	**74**

Table 6: The performances of the supervised Tw-StAR for TSAC dataset with preprocessing

Preprocessing	**Features**	**Model**	**P.(%)**	**R.(%)**	**F1.(%)**	**Acc.(%)**
	doc embeddings	MLP [11]	78.0	78.0	78.0	78.0
Stop	uni	SVM	92.5	92.3	92.4	92.6
Stem	uni	SVM	93.4	93.4	93.4	93.5
Neg	uni	SVM	92.6	92.5	92.5	92.7
Emo	uni	SVM	92.4	92.39	92.4	92.5
Stem + Stop	uni	SVM	93.8	93.8	93.8	93.9
Emo + Stop	uni	SVM	92.1	92.1	92.2	92.3
Emo + Stem	uni	SVM	**93.9**	**93.8**	**93.9**	**94.0**
Emo + Neg	uni	SVM	92.5	92.4	92.5	92.6
Emo + Stem + Stop	uni	SVM	93.8	93.8	93.8	93.9
Emo+ Stem + NEs	uni	SVM	92.8	92.86	92.8	93

Table 7: The performances of lexicon-based Tw-StAR in TSAC dataset with preprocessing

Preprocessing	**Features**	**Model**	**P.(%)**	**R.(%)**	**F1.(%)**	**Acc.(%)**
	doc embeddings	MLP [11]	78	78	78	78
Stop	uni+bi	Lex	82	69	68.3	73.2
Stem	uni+bi	Lex	82.6	72	72	75.6
Neg	uni+bi	Lex	82.8	69.5	68.8	73.6
Emo	uni+bi	Lex	82.5	70.9	70.6	74.4
Stem + Stop	uni+bi	Lex	81.7	71.78	71.7	75.3
Emo + Stop	uni+bi	Lex	82.3	70.4	70	74.3
Emo + Stem	uni+bi	Lex	83	73	73.2	76.5
Emo + Neg	uni+bi	Lex	83.1	70.8	70.5	74.7
Emo + Stem + Stop	uni+bi	Lex	82.2	72.9	73.1	76.3
Emo+ Stem + NEs	uni+bi	Lex	**83.2**	**83.4**	**81.9**	**81.9**

Emoji were detected only in TSAC dataset as TEC and TAC datasets do not contain any emoji. In TSAC, emoji tagging had no significant impact on the performance when it was separately applied whereas combining emoji tagging along with stemming scored the best F1-score among all the experiments with a value equals to 93.9%. In the lexicon-based model (see Table 7), however, emoji tagging could not outperform the baseline either when it was applied solely or combined with negation tagging task. Moreover, applying emoji tagging together with negation in the supervised model, achieved almost the same results scored by the negation preprocessing task. This could be due to sarcastic content in which emoji do not express the true meant sentiment but its opposite.

Considering Tables 3, 5 and 7, it is clear that the accuracy in all datasets decreased when negation tagging was conducted in the lexicon-based model. On the other hand, the precision was increased while the recall was decreased. This indicates that with negation detection applied, tweets were classified more accurately but the number of the classified instances was low. As for the supervised model, Tables 2, 4 and 6 show that the performances were improved in all datasets when negation detection was applied. Nevertheless, the least improvement was reported in TEC as the accuracy was increased by 2.31% compared to 13.9% and 14.7% increment ratios scored in TAC and TSAC datasets respectively. This could be attributed to the ability of the proposed negation detection strategy to capture negations in datasets of pure Tunisian content (TAC, TSAC) more accurately than those of mixed Tunisian/MSA content such as TEC.

The lexicon-based performances listed in Table 3, Table 5 and Table 7 emphasize the role of NEs in improving SA performance. Combining NEs with negation and stemming scored the best performances where the baseline was outperformed in all datasets. For instance, in TSAC dataset when NEs were merged with stemming and emoji tagging, an accuracy of 81.9% was recorded compared to 78% scored by the baseline system [11]. Moreover, in the supervised model, tagging NEs along with negation in TAC and with stemming in TEC improved the F1-score by 6.7% and 4.8% in TAC and TEC datasets respectively.

6 Conclusion and Future Work

This study has shed light on the vital role of preprocessing phase in sentiment analysis of the Tunisian Dialect. Examining various preprocessing tasks with supervised and lexicon-based models specified stemming, emoji and negation tagging as the most effective tasks for Tunisian SA. Hence, combining the novel preprocessing task NEs tagging with these effective tasks led to the best SA performances as the baselines were outperformed by a significant margin. For the future work, SA performances would be further improved if negation detection strategy was extended to handle irony and sarcastic content. In addition, using a Tunisian stopwords list instead of the MSA stopwords adopted here might enhance the stemming task. Finally, it would be better if Tunisian corpora were provided to produce the pretrained word vectors used in NER system such that special Tunisian NEs such as the singer name “كافون” could be recognized as a person name and tagged properly, rather than being identified as the MSA word that means “enough” and stemmed into “كاف”.

References

1. **Abdelali, A., Darwish, K., Durrani, N., & Mubarak, H. (2016).** Farasa: A fast and furious segmenter for arabic. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pp. 11–16.
2. **Akaichi, J. (2014).** Sentiment classification at the time of the tunisian uprising: Machine learning techniques applied to a new corpus for arabic language. *Network Intelligence Conference (ENIC), 2014 European*, IEEE, pp. 38–45.
3. **Aly, M. & Atiya, A. (2013).** Labr: A large scale arabic book reviews dataset. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pp. 494–498.

4. **Brahimi, B., Touahria, M., & Tari, A. (2016).** Data and text mining techniques for classifying arabic tweet polarity. *Journal of Digital Information Management*, Vol. 14, No. 1.

5. **Duwairi, R. & El-Orfali, M. (2014).** A study of the effects of preprocessing strategies on sentiment analysis for arabic text. *Journal of Information Science*, Vol. 40, No. 4, pp. 501–513.

6. **El-Beltagy, S. R., El kalamawy, M., & Soliman, A. B. (2017).** Niletmrg at semeval-2017 task 4: Arabic sentiment analysis. *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, Association for Computational Linguistics, pp. 790–795.

7. **Gridach, M. (2016).** Character-aware neural networks for arabic named entity recognition for social media. *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP2016)*, pp. 23–32.

8. **Karmani, N. (2017).** *Tunisian Arabic Customer's Reviews Processing and Analysis for an Internet Supervision System*. PhD dissertation, Sfax University, Tunisia.

9. **Larkey, L. S., Ballesteros, L., & Connell, M. E. (2002).** Improving stemming for arabic information retrieval: light stemming and co-occurrence analysis. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, ACM, pp. 275–282.

10. **Malmasi, S., Refaee, E., & Dras, M. (2015).** Arabic dialect identification using a parallel multidialectal corpus. *International Conference of the Pacific Association for Computational Linguistics*, Springer, pp. 35–53.

11. **Medhaffar, S., Bougares, F., Esteve, Y., & Hadrich-Belguith, L. (2017).** Sentiment analysis of tunisian dialects: Linguistic ressources and experiments. *Proceedings of the Third Arabic Natural Language Processing Workshop*, pp. 55–61.

12. **Mulki, H., Haddad, H., & Gridach, M. (2017).** Polarity analysis of non figurative tweets: Tw-star participation on deft 2017. *24e Conférence sur le Traitement Automatique des Langues Naturelles (TALN)*, pp. 92–98.

13. **Mulki, H., Haddad, H., Gridach, M., & Babaoğlu, I. (2017).** Tw-star at semeval-2017 task 4: Sentiment classification of arabic tweets. *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 664–669.

14. **Piryani, R., Madhavi, D., & Singh, V. K. (2017).** Analytical mapping of opinion mining and sentiment analysis research during 2000–2015. *Information Processing & Management*, Vol. 53, No. 1, pp. 122–150.

15. **Rosenthal, S., Farra, N., & Nakov, P. (2017).** Semeval-2017 task 4: Sentiment analysis in twitter. *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 502–518.

16. **Rushdi-Saleh, M., Martín-Valdivia, M. T., Ureña-López, L. A., & Perea-Ortega, J. M. (2011).** Oca: Opinion corpus for arabic. *Journal of the Association for Information Science and Technology*, Vol. 62, No. 10, pp. 2045–2054.

17. **Samih, Y., Attia, M., Eldesouki, M., Abdelali, A., Mubarak, H., Kallmeyer, L., & Darwish, K. (2017).** A neural architecture for dialectal arabic segmentation. *Proceedings of the Third Arabic Natural Language Processing Workshop*, pp. 46–54.

18. **Sayadi, K., Liwicki, M., Ingold, R., & Bui, M. (2016).** Tunisian dialect and modern standard arabic dataset for sentiment analysis: Tunisian election context.

19. **Yasavur, U., Travieso, J., Lisetti, C. L., & Rishe, N. D. (2014).** Sentiment analysis using dependency trees and named-entities. *Proceedings of the Twenty-Seventh International Florida Artificial Intelligence Research Society Conference*, pp. 134–139.

Article received on 09/12/2017; accepted on 15/02/2018.
Corresponding author is Hala Mulki.