The UNIBA System at the EVALITA 2018 Italian Emoji Prediction Task

Lucia Siciliani and **Daniela Girardi**
Department of Computer Science, University of Bari Aldo Moro
Via, E. Orabona, 4 - 70125 Bari (Italy)
{lucia.siciliani,daniela.girardi}@uniba.it

Abstract

English. This paper describes our participation in the ITAmoji task at EVALITA 2018 (Ronzano et al., 2018). Our approach is based on three sets of features, i.e. micro-blog and keyword features, sentiment lexicon features and semantic features. We exploit these features to train and combine several classifiers using different libraries. The results show how the selected features are not appropriate for training a linear classifier to properly address the emoji prediction task.

Italiano. *Questo articolo descrive l'approccio utilizzato per la partecipazione al task ITAmoji di EVALITA 2018 (Ronzano et al., 2018). Il nostro metodo si basa su tre insiemi di features: il primo rappresenta le informazioni intrinseche dei messaggi all'interno dei microblog, il secondo riguarda le informazioni derivanti dal lessico ed infine un terzo creato usando i principi di semantica distribuzionale. Queste features sono state utilizzate per addestrare diversi classificatori attraverso diverse librerie. I risultati ottenuti mostrano come le features selezionate non sono appropriate per addestrare un classificatore lineare nel task di predizione delle emoji.*

1 Introduction

Nowadays, emojis are widely used to express sentiments and emotions in written communication, which is becoming more and more popular due to the increasing use of social media. In fact, emojis can help the user to express and codify many different messages which can be also easily interpreted by a great audience since they are very intuitive. However, sometimes happens that their meaning is misleading, resulting in the misunderstanding of the entire message. The emoji detection has captured the interest of research since they could be relevant to improve sentiment analysis and user profiling tasks as well as the retrieval of social network material.

In particular, in the context of the International Workshop on Semantic Evaluation (SemEVAL 2018), the Multilingual Emoji Prediction Task (Barbieri et al., 2018) has been proposed for challenging the research community to automatically model the semantics of emojis occurring in English and Spanish Twitter messages. During this challenge, (Barbieri et al., 2017) created a model which outperforms humans in predicting the most probable emoji associated with a given tweet.

Twitter supports more than 1.000 emojis[1], belonging to different categories (e.g.: smiley and people, animals, fruits, etc.) and this number seems to grow.

In this paper, we used a set of features which showed promising results in predicting sentiment polarity in tweets (Basile and Novielli, 2014) in order to understand whether they could be used also to predict emoji or not. The paper is organized as follow: Section 2 describes the system and the exploited features, while in Section 3 we report the obtained results using different classifiers and their ensemble. Finally, in Section 4 we discuss our findings and Section 5 reports the conclusions.

2 System Description

In this section, we describe the approach used for solving the ITAmoji challenge. This task is structured as a multi-class classification since for each tweet it is possible to assign one of 25 emoji which however are mutually exclusive.

[1] https://it.piliapp.com/twitter-symbols/

The feature extraction was performed entirely using the language Java. First of all, each tweet was tokenized and stop-words were removed exploiting the "Twitter NLP and Part-of-Speech Tagging" API[2] developed by the Carnegie Mellon University. No other NLP steps, like stemming or PoS-tagging where considered since those features were considered not relevant for this particular kind of task.

Then we moved to the extraction of the features from the training data. These features can be categorized into three sets: one addressing the keywords and micro-blog features, the second one exploiting the polarity of each word in a semantic lexicon and the third one using their representation obtained through a distributional semantic model. A description of the different sets of features will be provided in Section 2.1.

After the features extraction, we obtained a total set of 342 features to be used to train a linear classifier. For classification, we decided to exploit the Weka API[3] and use an ensemble of three different classifiers to obtain better predictive results. The three classifiers that have been used are: the L2-regularized_L2-loss_support_vector_classification, the L2-regularized_logistic_regression, and the random forest classifier. The first two algorithms are based on the WEKA wrapper class for the Liblinear classifier (Fan et al., 2008) and were trained on the whole set of features, while the random forest was trained only over the keyword and micro-blog features. All the classifiers were combined using the soft-voting technique, which averages the sum of the output of each classifier over their overall number.

In the light of the results of the task given by the organizers, we conducted an in-depth analysis of our solution and discovered that due to a problem in the Liblinear WEKA wrapper, not all the classifiers returned a set of probability scores for multi-class classification thus compromising the results of all the ensemble. Therefore, even if out of the time scope of this challenge, we decided to try to use the scikit-learn (Buitinck et al., 2013) to build our classifiers and evaluate the impact of the selected features.

All the results will be summarized and discussed in Section 3 and Section 4.

[2]http://www.cs.cmu.edu/ ark/TweetNLP/

[3]9http://www.cs.waikato.ac.nz/ml/weka/

2.1 Features

As in the previous work of (Basile and Novielli, 2014), we defined three groups of features based on (i) keyword and micro-blogging characteristics, (ii) a sentiment lexicon and (iii) a Distributional Semantic Model (DSM). Keyword based features exploit tokens occurring in the tweets, considering only unigrams. During the tokenization phase user mentions, URLs and hash-tags are replaced with three meta-tokens: "USER", "URL", and "TAG", in order to count them and include their number as features. Other features connected to the micro-blogging environment are: the presence of exclamation and interrogative marks, adversative, disjunctive, conclusive, and explicative words, the use of uppercase and informal expressions of laughter, such as "ah ah". The list of micro-blogging features is reported in 1.

The second block of features consists of sentiment lexicon features. As Italian lexicon database, we used MultiWordNet (Pianta et al., 2002), where at each lemma is assigned a positive, negative and neutral score. In particular, we include features based on the prior polarity of words in the tweets. To deal with mixed polarity cases we defined two sentiment variation features so as to capture the simultaneous expression of positive and negative sentiment. We decided to include features related to the polarity of the tweets since emoji could be intuitively categorized into positive and negative and are usually used to enforce the sentiment expressed. The list of sentiment lexicon features is reported in 2. The last group of features is the semantic one, which exploits a Distributional Semantic Model. We used the vector embeddings for each word and the superposition operator (Smolensky, 1990) to compute an overall vector representation of the tweet. Analogously, we first computed a prototype vector for each polarity class (positive, negative, subjectivity and objectivity) as the sum of all the vector representations of each tweet to a certain class. Finally, we computed the element-wise minimum and maximum of the vectors representation of each word in the tweet and then the resulting vectors were then concatenated and used as features. This approach has been proved to work well and easy to compute for small texts like tweets and other micro-blog posts (De Boom et al., 2016). The list of sentiment lexicon features is reported in 3.

Microblog	Description
tag	total occurences of hashtags
url	total occurrences of URLs
user	total occurrences of user mentions
neg_count	total occurrences of ”non” word pt
exclamation	total occurrences of exclamation marks
interrogative	total occurrences of interrogative marks
adversative	total occurrences of adversative words
disgiuntive	total occurrences of disjunctive words
conclusive	total occurrences of conclusive words
esplicative	total occurrences of esplicative words
uppercase_ch	number of upper case characters
repeat_ch	number of consecutive repetitions of a character in a word
ahah_repetition	total occurrences of ”ahah” laughter expression

Table 1: Microblog Features.

3 Evaluation

The goal of the ITAmoji challenge is to evaluate the capability of each system to predict the right emoji associated with a tweet, regardless of its position in the text.

Organizers selected a subset of 25 emojis and provided 250,000 tweets for training, each tweet contains only one emoji which is extracted from text and given as a target feature. The training set is very unbalanced since three emojis (i.e.: read_heart ❤️, face_with_tears_of_joy 😂, and smiling_face_with_heart_eyes 😍) represent almost 50% of the whole dataset.

For the evaluation instead, the organizers created a test set made up of 25,000 tweets, keeping unchanged the ratio of the different classes over the whole set. The prediction for each tweet is composed by the list of all the 25 emojis ordered by their probability to be associated to the tweet: in this way, it is possible to evaluate the systems according to their accuracy up to a certain position in the rank. Nevertheless only the first emoji one was mandatory for the submission.

Systems were ranked according to the macro F-Measure but also other metrics have been calculated, i.e. the micro F-measure, the weighted F-measure, the coverage error and the accuracy (measured @5, @10, @15 and @20). The final results for the challenge are reported in table 5. We can see how while there is quite a difference between the results obtained for the macro-F1 score, the same does not happen with the micro F1 score. The same happens with the outcomes of the accuracy where, setting aside two runs, all the other obtain a result which is included between 0,5 and 0,8. In other words, even if the macro-F1 measure appears to be the most discriminating factor among all the runs, such a result is based on the presence of some classes which appear over a numerous amount of instances and this causes the classifiers to overfit over them.

Table 6 summarizes the results obtained using both WEKA (the one which was submitted, highlighted in italic) and scikit-learn. We used the scikit-learn library to perform a classification using the logistic regression and then adding, using a soft voting technique, a Naive-Bayes classifier and a Random Forest (rows 4 and 5 respectively). From these results we can see how, independently from the used classifier, the final results in terms of the metrics used for the evaluation over the test dataset stay quite similar among them. Specifically, these results depends on the fact that our system predicts only two label as first which are ”red_heart” and ”face whit tears”, resulting unable to classify correctly the other classes, as is shown in table 4. This outcome is then probably due to the set of features that we used, which does not manage to appropriately model the data in this task, even if it proved to be successful in another sentiment analysis context (Basile and Novielli, 2014). In the last column of table 6, we reported the average macro-F1 obtained performing 5-fold cross validation. The value for the first evaluation has not been calculated since the fault in the library described in section 2.

Sentiment Lexicon	Description
subjScore	sum of the positive and negative scores
objScore	sum of the neutral scores
hitSubj	number of tokens having the positive or negative score higher than zero
hitObj	number of tokens having the neutral score higher than zero
avgSubj	the ratio between subScore/hitSubj
avgObj	the ratio between objScore/hitObj
subObjDiff	difference
posScore	sum of positive scores for the tokens in the tweet
negScore	sum of negative scores for the tokens in the tweet
hitPos	number of tokens that have the positive score higher than zero
hitNeg	number of tokens that have the negative score higher than zero the
avgPos	ratio between posScore and hitPos
avgNeg	ratio between negScore and hitNeg
posnegScore	difference between avgPos and avgNeg
max_sum_subj_ratio	ratio between the maximum subjScore and number of token having positive and negative score higher than zero
max_obj_score_ratio	ratio between the maximum objScore and number of token having neutral score higher than zero
avgMaxPos	ratio between maxSumPos and hitMaxPos
avgMaxNeg	ration between maxSumNeg and hitMaxNeg
diff_avg_max_pos_neg	difference between avgMaxPos and avgMaxNeg
sentiment_variation	for each token occurring in the tweet a tag is assigned, according to the highest polarity score of the token in the Italian lexicon Tag values are in the set OBJ, POS, NEG The sentiment variation counts how many switches from POS to NEG, or vice versa, occur in the tweet
sentiment_variation_posneg	it is similar to the previous feature, but the OBJ tag is assigned only if both positive and negative scores are zero. Otherwise, the POS tag is assigned f the positive score is higher than the negative one, vice versa the NEG tag is assigned.
intensity	intensity of the tweet
polarity	polarity of the tweet

Table 2: Sentiment Lexicon Features.

4 Discussion

The overall results of the challenge show how this task is non-trivial and difficult to solve with high precision and the reason behind this is intrinsic to the task itself. First of all, there are several emojis which often differ only slightly from each other, furthermore, this meaning is deeply dependent on the single user and from the context. In fact, a single emoji (like 😂) could be used to convey both joy and fun or, on the contrary, it could also be used ironically with a negative meaning. To this extent, an interesting update for the task could be to leave the text of the tweet as it is so that the position could be also exploited to detect irony and other variations.

From the analysis of the overall results of the task emerged that there is a large gap between the macro-F1 scores which is not reflected by the micro-F1. For this particular task, where both training and testing dataset are heavily unbalanced, we think that the micro-F1 score is more suited to capture the performance of the submitted systems since it takes into account the support of each class.

There is a result which is particularly interesting that is, the value for the 5-fold using only the logistic regression as a classifier which is particularly high (0,358) and is opposing to the final score. This aspect surely needs further investiga-

Semantic	Description
vec	the sum of the vector representations of each word in the tweet
simNeg	the similarity between $\vec{t}$ and the negative prototype vector $\vec{p_s}$
simPos	the similarity between $\vec{t}$ and the positive prototype vector $\vec{p_s}$
simSubj	the similarity between $\vec{t}$ and the subjective prototype vector $\vec{p_s}$
simObj	the similarity between $\vec{t}$ and the objective prototype vector $\vec{p_s}$
vecMin	the element-wise minimum of the vectors representations of each word in the tweet
vecMax	the element-wise maximum of the vectors representations of each word in the tweet

Table 3: Semantic Features.

tions.

5 Conclusion

In this paper, we presented our contribution to the ITAmoji task of the EVALITA 2018 campaign. We tried to model the data by extracting features based on the keywords and micro-blogging characteristics, using a sentiment lexicon and finally using word embeddings. Apart from the characteristics of the different libraries available for machine learning purposes, the results show how, independently from the classifier, those features do not adapt to this problem. As future work, this analysis could also be extended with an ablation which would allow understanding if there are noisy features.

References

Francesco Barbieri, Miguel Ballesteros, and Horacio Saggion. 2017. Are emojis predictable? *arXiv preprint arXiv:1702.07285*.

Francesco Barbieri, Jose Camacho-Collados, Francesco Ronzano, Luis Espinosa Anke, Miguel Ballesteros, Valerio Basile, Viviana Patti, and Horacio Saggion. 2018. Semeval 2018 task 2: Multilingual emoji prediction. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 24–33.

Pierpaolo Basile and Nicole Novielli. 2014. Uniba at evalita 2014-sentipolc task: Predicting tweet sentiment polarity combining micro-blogging, lexicon and semantic features. *Proceedings of EVALITA*, pages 58–63.

Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Cedric De Boom, Steven Van Canneyt, Thomas Demeester, and Bart Dhoedt. 2016. Representation learning for very short texts using weighted word embedding aggregation. *Pattern Recognition Letters*, 80:150–156.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *Journal of machine learning research*, 9(Aug):1871–1874.

Emanuele Pianta, Luisa Bentivogli, and Christian Girardi. 2002. Multiwordnet: developing an aligned multilingual database. 1st gwc. *India, January*.

Francesco Ronzano, Francesco Barbieri, Endang Wahyu Pamungkas, Viviana Patti, and Francesca Chiusaroli. 2018. Overview of the EVALITA 2018 Italian Emoji Prediction (ITAMoji) Task. In *Proceedings of Sixth Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2018)*, Turin, Italy. CEUR.org.

Paul Smolensky. 1990. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial intelligence*, 46(1-2):159–216.

label	precision	recall	f1-score	support
beaming_face_with_smiling_eyes	0,000	0,000	0,000	1028
blue_heart	0,500	0,002	0,004	506
face_blowing_a_kiss	0,500	0,002	0,005	834
face_savoring_food	0,000	0,000	0,000	387
face_screaming_in_fear	0,000	0,000	0,000	444
face_with_tears_of_joy	0,313	0,448	0,369	4966
flexed_biceps	0,000	0,000	0,000	417
grinning_face	0,000	0,000	0,000	885
grinning_face_with_sweat	0,000	0,000	0,000	379
kiss_mark	0,000	0,000	0,000	279
loudly_crying_face	0,000	0,000	0,000	373
red_heart	0,259	0,909	0,403	5069
rolling_on_the_floor_laughing	0,000	0,000	0,000	546
rose	0,125	0,004	0,007	265
smiling_face_with_heart_eyes	0,135	0,004	0,008	2363
smiling_face_with_smiling_eyes	0,167	0,000	0,002	1282
smiling_face_with_sunglasses	0,000	0,000	0,000	700
sparkles	0,000	0,000	0,000	266
sun	0,000	0,000	0,000	319
thinking_face	0,000	0,000	0,000	541
thumbs_up	0,000	0,000	0,000	642
top_arrow	0,000	0,000	0,000	347
two_hearts	0,000	0,000	0,000	341
winking_face	0,000	0,000	0,000	1338
winking_face_with_tongue	0,000	0,000	0,000	483
avg / total	**0,164**	**0,274**	**0,156**	**25000**

Table 4: Classification report for each class.

teamName	macroF1	microF1	weightedF1	covErr	acc@5	acc@10	acc@t15	acc@t20
FBK_FLEXED	0,365	0,477	0,470	3,470	0,817	0,921	0,969	0,991
FBK_FLEXED	0,356	0,476	0,466	3,486	0,815	0,919	0,968	0,992
FBK_FLEXED	0,292	0,423	0,396	4,354	0,745	0,875	0,943	0,980
GW2017	0,233	0,401	0,378	5,662	0,672	0,815	0,894	0,930
GW2017	0,222	0,422	0,369	4,601	0,713	0,859	0,943	0,983
CIML-UNIPI	0,192	0,291	0,315	5,432	0,646	0,830	0,930	0,980
CIML-UNIPI	0,188	0,376	0,341	5,114	0,685	0,839	0,924	0,973
sentim	0,106	0,294	0,232	6,412	0,585	0,769	0,885	0,957
sentim	0,102	0,313	0,231	6,326	0,576	0,772	0,897	0,964
GW2017	0,038	0,119	0,110	13,489	0,279	0,430	0,560	0,663
UNIBA	0,032	0,274	0,156	6,697	0,588	0,760	0,864	0,935
sentim	0,019	0,065	0,040	12,458	0,292	0,488	0,644	0,740

Table 5: Final results of the challenge.

runName	macroF1	microF1	weightedF1	covErr	acc@t5	acc@10	acc@15	acc@20	K-fold
UNIBA_weka	*0,032*	*0,274*	*0,156*	*6,697*	*0,588*	*0,760*	*0,864*	*0,935*	-
UNIBA_sklearn_lr	0,039	0,257	0,156	6,459	0,610	0,765	0,873	0,947	0,358
UNIBA_sklearn_lr_nb	0,032	0,195	0,119	6,634	0,604	0,761	0,868	0,946	0,120
UNIBA_sklearn_lr_rf_nb	0,032	0,214	0,126	6,749	0,582	0,758	0,869	0,946	0,183

Table 6: Evaluation of the other classifiers using the same set of feature. In the second row are reported the results of our first submission. The last column reports the average macroF1 obtained performing a K-fold cross validation.