Hatching Chick at SemEval-2018 Task 2: Multilingual Emoji Prediction

J. Coster, R.G. van Dalen, and **N.A.J. Stierman**
Department of Information Science
University of Groningen
`{j.coster.2, r.g.van.dalen, n.a.j.stierman}@student.rug.nl`

Abstract

As part of a SemEval 2018 shared task an attempt was made to build a system capable of predicting the occurence of a language's most frequently used emoji in Tweets. Specifically, models for English and Spanish data were created and trained on 500.000 and 100.000 tweets respectively. In order to create these models, first a logistic regressor, a sequential LSTM, a random forest regressor and a SVM were tested. The latter was found to perform best and therefore optimized individually for both languages. During developmet f1-scores of 61 and 82 were obtained for English and Spanish data respectively, in comparison, f1-scores on the official evaluation data were 21 and 18. The significant decrease in performance during evaluation might be explained by overfitting during development and might therefore have partially be prevented by using cross-validation. Over all, emoji which occur in a very specific context such as a Christmas tree were found to be most predictable.

1 Introduction

It is said that a picture is worth a thousand words; inherently then, visual icons can provide additional meaning to text. One common example of this is the use of emoticons (emoji) accompanying primarily short, informal texts such as text messages and tweets. The role of these is interesting as they can be used in a variety of manners such as to complement text (e.g., 😊 to indicate happiness) and to replace text (e.g., *I* ❤ *you* instead of *I love you*).

This research is concerned with the development of a system for predicting the occurrence of these emoji on the basis of Twitter data and is conducted as part of a *SemEval 2018* shared task (Barbieri et al., 2018). More specifically, given the text of an English or Spanish tweet, the system will attempt to predict the emoji originally found in that tweet. The set of emoji used is limited to the twenty most frequent emoji for each language respectively.

2 Previous Work

A similar research on predicting emoji occurring in tweets was conducted by Barbieri et al. (2017). In this study a Bidirectional Long Short-Term Memory (BLSTM) network with standard lookup word representations and character-based representations of tokens was used. While it was found that profoundly dissimilar emoji could be predicted, this method did not succeed to accurately differentiate between twenty emoji classes.

Na'aman et al. (2017) meanwhile conducted a study to further explore the linguistic varieties of the purposes of emoji on Twitter. It was found that emoji can be an integral part of the content. One common example of this is a part of a phrase being replaced by an emoji, much like the '*I* ❤ *you*' example from the previous section. The notion of these word-emoji combinations is also mentioned by Dimson (2015). These findings inherently strongly support the idea that text could to some extent be used to predict emoji, given that fixed co-occurrences do exist.

3 Data

Since this study was conducted as part of a shared task, data was made available by the organization (Barbieri et al., 2018). For the English language, data consisted of a trial set containing 50.000 and a training set containing 500.000 tweets geolocated in the United States. For Spanish, a trial set of 10.000 and a training set of 100.000 tweets geolocated in Spain were available. For both languages, the trial portion of the data was used as a development set. Evaluation then was ultimately conducted on a held out data set sized similarly to the

Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), pages 445–448
New Orleans, Louisiana, June 5–6, 2018. ©2018 Association for Computational Linguistics

	English			Spanish	
	Train	Test		Train	Test
❤️	118425	10798	❤️	21854	2028
😍	57167	4830	😍	14962	1363
😂	56199	4534	😂	10299	970
💕	30360	2605	💕	7526	705
🔥	27137	3716	😊	7137	645
😊	25614	1613	😘	4834	415
😎	23362	1996	💪	4220	367
✨	20259	2749	😉	4053	386
💙	18817	1549	👌	3857	320
😘	17828	1175	🇪🇸	3776	369
📷	17586	1423	😎	3420	267
🇺🇸	16876	1949	💙	3198	271
☀️	15333	1265	💜	3112	313
💜	14323	1114	😜	2993	281
😉	14842	1306	💞	2908	282
💯	14655	1244	✨	2806	244
😁	14394	1153	🎶	2861	262
🎄	14122	1545	💘	2785	260
📸	14534	2417	😁	2807	252
😜	13516	1010	-	-	-

Table 1: The distribution of training and test data as used during evaluation

trial set for both languages respectively.

Analysis of the supplied data showed the tweets originated from the period between October 2015 and February 2017 with a natural distribution to the extent that slightly more tweets originated from months with a large number of public holidays such as December. All of these tweets contain one of the twenty most used emoji for their respective language, although due to an error in the data ultimately only the top nineteen had to be predicted for Spanish. The distribution of the emoji over the tweets can be seen in Table 1. Note that in this table counts for the trial and training data have been merged as a data set combining these was used during final evaluation.

4 Method

In order to build capable models for predicting emoji in English and Spanish tweets, a two step procedure was followed. First, various forms of preprocessing and multiple machine learning algorithms were tested in order to identify what type of model would most likely be successful. Then, a system built on these results was optimized for English and Spanish tweets separately in order to create a model for each language.

4.1 Initial Model Selection

In order to determine which type of model to use, four different classification methods were tested. Specifically, a *Logistic Regression* classifier was tested using word unigrams, word bigrams and a combination of both. Next, a sequential LSTM with maximum sentence length embeddings as features was tested. This model used 40 neurons and a *softmax* activation function. The model was compiled by implementing *Categorical Crossentropy* and the *Adagrad* optimizer. Then, a *Random Forest Regressor* was implemented using the same features as the initial *Logistic Regression* model. Finally, a *linear SVM* model was built using the SKLearn *SGDClassifier* with a hinge loss function.

For all models, four preprocessing steps were tested. Namely replacing URLs occurring in the data with a general identifier, replacing mentions occurring in the data with a general identifier, tokenizing the tweets using the *NLTK TweetTokenizer* and stemming the tweets using the *Snowball* stemmer.

After all tests had been executed it was found that the linear SVM using the SGDClassifier yielded the best results. Therefore this model was selected as a basis for the per language models.

4.2 English

4.2.1 Preprocessing

For the optimized English model, most of the preprocessing steps from the previous subsection; tokenization, URL replacement and mention replacement, were used. Use of the snowball stemmer was omitted as it did not appear to improve performance. Additionally, punctuation was removed as this seemed to yield better results on trial data.

4.2.2 Optimized Model

As set out in the *Initial Model Selection* subsection the SKLearn *SGDClassifier* was used as the basis for this model. The settings used for this classifier can be seen in Table 2 and are shared with the model for Spanish. The input for this classifier then was a tf-idf vector created from the preprocessed data, which was first converted to lowercase as this was found to improve performance.

The SKLearn *FeatureUnion* function was used in order to experiment with both word and character ngram ranges simultaneously. Ultimately it was found that using only word ngrams with a range of two to four yielded the best results. However, despite not using character ngrams in the optimized model, the *Featurenion* was kept as the 0.5 weight it applied to all features improved results by approximately two percent point. This effect is most likely caused by the reduction of the absolute differences between the predictiveness of features. After these optimizations, testing on the trial set resulted in an average f1-score of 61.

4.3 Spanish

4.3.1 Preprocessing

For the final model, the Spanish optimized model used the same preprocessing procedure as the English optimized model, as described in section 4.2.1 as this procedure was found to perform best on Spanish data as well. During the development of this model however the use of a lemmatizer at the preprocessing stage was also tested as a replacement of the Snowball stemmer. While neither were included in the final model as they did not yield a significant improvement, it is interesting to note that the model with lemmatizer scored better when its language was set to English as opposed to Spanish, despite the language of the data primarily being the latter.

4.3.2 Optimized Model

Much like the English optimized model, the final model for Spanish data used the SKLearn SGDClassifier with the same parameter settings, as seen in Table 2. The only difference then is that for Spanish data using a ngram range of one to seven instead of two to four was found to yield the best results. When tested against the trial data this model yielded an average f1-score of 82.

5 Results

Once parameter optimization on both the English and Spanish models had been completed, the models were prepared for official evaluation on previously unseen data as explained in section 3. To this end, a merged data set containing both training and trial data was created for each language respectively as during development it was found that system performance would scale with the amount of

Parameter	**Value**
loss	hinge
penalty	l2
alpha	1e-3
random_state	42
max_iter	20
tol	None
class_weight	dict(*1 for each class*)

Table 2: Parameters used for the *SGDClassifier*

English		**Spanish**	
Emoji	**F1-score**	**Emoji**	**F1-score**
❤️	57.29	❤️	64.725
😍	26.796	😍	34.635
😂	37.755	😂	51.356
💕	7.931	💕	6.847
🔥	41.762	😊	10.506
😊	7.11	😘	20.755
😎	12.034	💪	32.701
✨	16.7	😉	9.339
💙	9.122	👌	10.631
😘	5.9	🇪🇸	44.649
📷	14.359	😎	11.268
🇺🇸	52.366	💙	6.391
☀️	33.295	💜	1.439
💜	5.459	😜	4
😉	5.472	💞	4.651
💯	12.513	✨	13.843
😁	3.254	🎶	21.277
🎄	57.807	💘	6.03
📸	19.568	😁	0.806
😜	2.26	-	-
Average	**21.438**	**Average**	**18.729**

Table 3: F1-scores achieved during evaluation

training data used[1]. The models were then trained on these merged data sets and tasked with predicting the corresponding emoji for the tweets in the evaluation data. These predictions were submitted to and consecutively evaluated by the task's organization. Results from this evaluation are detailed in the following subsections.

5.1 English Model

On average, the English optimized classifier achieved a f1-score of 21.438 with a precision of 25.965, a recall of 21.483 and an accuracy of 36.522. An overview of per class performance in the form of f1-scores can be seen in Table 3. Overall, the system performed best when predicting emoji which are likely to only occur in a specific context. ❤️ For example is likely to occur in tweets about love, 🇺🇸 is predominantly used in the context of independence day and 🎄 is used mainly in tweets concerning Christmas. On these emoji the system achieved f1-scores of over 50. Meanwhile, the system performed worst on emoji such as 😊 and 😉, which are likely to be used in a plethora of different contexts. These findings are in line with trends seen when testing on trial data during development.

5.2 Spanish Model

Contrary to scores seen when testing on trial data, the Spanish optimized classifier performed slightly worse than the English optimized system. When tested on evaluation data, this model achieved a f1-score of 18.729 with a precision of 20.662, a recall of 19.163 and an accuracy of 37.23. Compared to English, a similar trend of weaker performance on more generic emoji is seen. Furthermore, ❤️ ranked among the most accurately predicted emoji for Spanish as well. However due to lack of knowledge of the Spanish language no qualitative analysis of why other emoji such as 😂 and 🇪🇸 could be predicted relatively well was conducted.

6 Discussion

Although compared to other systems participating in the task the models did not do exceptionally bad, a significant drop in performance is seen when compared to results obtained during development. In fact, the English model saw a 40 percent point drop and the Spanish model a 64 percent point drop. This decrease could partially be explained by differences in the distribution of certain emoji, as can be seen in Table 1. More importantly however, it is likely that the models were overfitted on the trial data as all testing during development was done on this portion of the data. In hindsight then, cross-validation might have been the better approach for evaluation during development.

Acknowledgments

Our contribution to this shared task was conducted as part of a course of the Information Science Master's degree programme at the University of Groningen. As such we would like to thank H. Haagsma for his insightful feedback during progress meetings.

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 (SemEval-2018)*, New Orleans, LA, United States. Association for Computational Linguistics.

Thomas Dimson. 2015. Emojineering part 1: Machine learning for emoji trends. *Instagram Engineering Blog*, 30.

Noa Na'aman, Hannah Provenza, and Orion Montoya. 2017. Varying linguistic purposes of emoji in (twitter) context. In *Proceedings of ACL 2017, Student Research Workshop*, pages 136–141.

[1] Debugging was often conducted using a portion of the training data in order to reduce execution time