Classifying the Informative Behaviour of Emoji in Microblogs

Giulia Donato*, **Patrizia Paggio***†
*University of Copenhagen
giulia.dnt@gmail.com, paggio@hum.ku.dk
†University of Malta
patrizia.paggio@um.edu.mt

Abstract

Emoji are pictographs commonly used in microblogs as emotion markers, but they can also represent a much wider range of concepts. Additionally, they may occur in different positions within a message (e.g. a tweet), appear in sequences or act as word substitute. Emoji must be considered necessary elements in the analysis and processing of user generated content, since they can either provide fundamental syntactic information, emphasize what is already expressed in the text, or carry meaning that cannot be inferred from the words alone. We collected and annotated a corpus of 2475 tweets pairs with the aim of analyzing and then classifying emoji use with respect to redundancy. The best classification model achieved an F-score of 0.7. In this paper we shortly present the corpus, and we describe the classification experiments, explain the predictive features adopted, discuss the problematic aspects of our approach and suggest future improvements.

Keywords: Emoji, Microblogs, Redundancy, Supervised Learning

1. Introduction

Emoji are non verbal features used to enrich computer mediated communication (CMC) and mobile mediated communication (MMC). Empirically, the use of emoji may vary in non trivial ways: emoji can be redundant with respect to the text, but they can also act as words, carrying their own part-of-speech category, thus providing fundamental semantic information. Studies on emotion expression in text have noted that emotional emoji may carry information which could not be inferred from the words alone. Following these findings, this work proposes to further investigate the informative behaviour of emoji in microblogs.

Recognizing to what extent emoji are redundant with respect to one or more words in the text could be helpful for further research in automatic content labeling; understanding how (and to what extent) emoji convey additional meaning or have a syntactic function can be important to improve the results in other NLP tasks such as metaphor detection and text summarization.

2. Literature

Emoji are picture characters, or pictographs, initially developed by Shigetaka Kurita during the late nineties. The initial set was further expanded and eventually became part of the Unicode standard in 2009 (Kelly, 2015; Miller et al., 2016).

Emoji in CMC are, similarly to emoticons, mostly used to express emotions; according to (Swiftkey, 2015) the top used emoji categories are the ones that include the happy and the sad faces. Novak et al. (2015) confirm that these preferences apply also to Twitter users. While Boia et al. (2013) demonstrate that emoticons are not necessarily accurate in retrieving sentiment words from a corpus, Hallsmar and Palm (2016) show that an emoji heuristic can be effectively used to retrieve tweets corresponding to a specific emotion class (within a multiclass framework), perhaps indicating that emotional emoji to some extent may behave redundantly with respect to the tweet text.

Since both emoticons and emoji can be employed with a wide range of purposes (Derks et al., 2007; Kelly and Watts, 2015), some research focused on their semantic aspects. Barbieri et al. (2016) analyzed the distribution of emoji in a corpus of more than 9.000.000 tweets. The emoji embeddings obtained were used to compute similarity and relatedness scores among emoji pairs and the results were evaluated by means of a manually annotated gold standard. The emoji vectors plotted in 2d space showed consistent semantic clusters, however the analysis of the words related to each cluster shows a noisy outcome. This suggests the need to observe the relation between emoji and words with different criteria, for example by considering how and how often emoji can represent information that is missing from the text. Eisner et al. (2016) elaborated on the findings in Barbieri et al. (2016); the authors demonstrated that emoji vectors generated from the sole Unicode descriptions are effective in classifying pairs of emoji with respect to their similarity.

Identifying redundant information in text is a useful step for text summarization or paraphrase detection. In microblogs repeated content can be found within specific conversations or topics. Zanzotto et al. (2011) provided a formal definition of linguistic redundancy in Twitter and performed machine learning experiments to quantify how common the phenomenon is in the social network. The authors experimented with several features combinations and found that the use of thesaurus metrics combined with partial syntactic analysis was the most effective in classifying redundant vs. non redundant instances. This paper is strongly inspired by the framework described in Zanzotto et al. (2011), and investigates the notion of redundant vs non-redundant behaviour in the domain of emoji use.

3. Methodology

The aim of this study is firstly to investigate how easy it is for human coders to distinguish different uses of emoji with respect to their semantic contribution, and secondly to experiment with automatic classification of emoji behaviour.

In order to explore both aspects, we set up a corpus of English tweets each of which contained at least one emoji. The corpus was annotated by four human coders in order to be used in a supervised machine learning experiment.

3.1. Corpus Creation

To collect the data we started by selecting a set of 30 emoji (three per each category among: *Nature & Animals, Places, Traveling/Commuting, Sport, Events, Other Activities, Music, Eating & Drinking, People, Feelings*) that we used as search keywords to retrieve relevant tweets automatically.
The raw data were cleaned and balanced (considering the category) and eventually sum up to a collection of 4100 tweet pairs. The annotation took place remotely between the 21st and the 31st of December 2016 and was performed by four annotators, three located in Greece and one in the Netherlands. All the annotators are fluent English speakers. We presented them with examples randomly drawn from the set of 4100 pairs. Each pair contained the same tweet twice: once with and once without a specific emoji. The coders were asked to select among three possible options to label each pair instance by considering the emoji of interest. The possible classes were: *Redundant*, *Non Redundant*, *Non Redundant + POS*; these classes were described and exemplified in ad-hoc instructions, on the basis of the definition of redundancy provided by Zanzotto et al. (2011).
The Redundant class indicates that the emoji of interest repeats the information present in the text or that its meaning is implied by the text. The Non-redundant class, on the contrary, captures cases in which the emoji adds information that is neither explicitly present nor implied in the text. Lastly the Non-Redundant + POS class, which refers to a specific kind of redundant use, indicates that the emoji is used with a syntactic function (and can be labeled with its own POS), thus replacing a word.
Examples to illustrate the three types of usage are listed below:

1. *Redundant*

   ”We’ll always have Beer. I’ll see to it. I got your back on that one. 🍺”

2. *Non-Redundant*

   ”I wish you were here ✈️”

3. *Non-Redundant + POS*

   ”Thank you so so so so much ily Here’s a 🍕 as a thank you gift x”

An more difficult case if the following one:

> ”Reading is always a good idea 📚. Thank you for your sincere support @USER. Happy reading.”

The content of the emoji might seem implied at first sight; however, books are not the only possible reading media, so in fact the emoji adds a more specific meaning to the tweet.

The annotators worked using an an interface specifically developed for the project. They had random access to roughly 1000 items each from the original sample of 4100 instances, and the process resulted in an annotated corpus consisting of 2475 unique pairs.
The analysis of the classes distribution shows the following counts: the *Redundant* class has 834 instances (33.7%), the *Non-Redundant* class has 1428 instances (57.7%), the *Non-Redundant + POS* class has 176 instances (7.1%). Additionally, 37 instances are annotated as *undefined* (1.5%).
The inter annotator agreement computed by means of the Cohen’s κ coefficient, shows a value of .56, which is considered to be moderate ((Landis and Koch, 1977)). This result and the difficulties reported by the annotators to assign a class in several cases, suggest that the task is not trivial for humans.
We additionally analyzed the corpus wrt potential features: in particular the position of the emoji (further described in 3.3.) and the emoji POS tag obtained from the Stanford POS Tagger were shown to be promising features to consider when training a classifier to predict whether emoji in tweets are being used in a redundant or additive way.
The creation of the corpus and its analysis is described in full in Donato and Paggio (2017).

3.2. Preprocessing

Before running the machine learning experiments, the corpus was preprocessed. Beside the standardization of mentions and links, the tweets were tokenized and stopwords and punctuation were removed. To achieve a proper tokenization on groups of emoji we used **Tweetokenize** (`https://github.com/jaredks/tweetokenize`), an online tokenizer designed for tweets tokenization. The tokenizer considers emoji, emoticons and hashtags as single tokens, and provides a series of further preprocessing methods such as lowercasing. Furthermore, the tokens were stemmed by means of the NLTK Wordnet lemmatizer. Since the Wordnet lemmatizer requires the indication of the word’s POS-tag to generate the stem, we ran a POS tagger on the tweets prior to their lemmatization. we adopted the standard Stanford POS Tagger from the Python NLTK wrapper since traditional POS taggers can achieve satisfactory results when compared with domain specific taggers (Derczynski et al., 2013) and since, to the best of our knowledge, Twitter-specific POS taggers do not provide tags for emoji.

3.3. Features

Emoji Position

Since, as also noted in Novak et al. (2015), the emotional emoji positioned towards the end of the tweet are the more emotionally loaded ones. we wanted to verify if the position any kind of emoji has in a tweet may be an indicator of a specific informative behaviour. For example if it is true that emoji are more likely to be put towards the end of the tweet when they repeat words in the text, the position of the emoji will be potentially interesting for a classification of the tweet with respect to emoji redundancy.

We analyzed this feature in our corpus, firstly by computing the percentage frequency distribution of two possible modalities (close to the end of the tweet or not). We observed that for the close to the end condition 35.7% of the instances are annotated as Redundant, 60.7% as Non-Redundant, 2.9% as Non-Redundant + POS, and 0.7% are undefined. In the opposite condition 31.6% instances are Redundant, 54.6% are Non-Redundant, 11.5% are Non-Redundant + POS, and 2.3% are undefined. We performed then a χ-squared test of independence we obtained (χ-squared = 81.644, df = 3, p-value < 0.001) which indicates a significant difference. An analysis of the residuals confirmed what emerged from the observation of the frequency distribution: the effect of position is highest in the case of the Non-Redundant + POS class.
We obtained the emoji position value by dividing the index of the emoji in the tokenized tweet by the number of tokens in the tweet. However, for the classification the position feature was encoded as binary (1 if the value is above 0.7, 0 otherwise).

Similarity Measure
To detect semantic relations between words we followed a thesaurus-based approach. Considering the WBOW model described in Zanzotto et al. (2011), we computed the distance between the vector of the tweet words (without the emoji under consideration) and the vector of the emoji description given by the Emojitracker integrated by the extended description present in the Unicode website.
To calculate the distances we relied on the Wu-Palmer similarity measure, and used WordNet as the external resource. The Wu-Palmer (*wup*) similarity measure is formalized as follows:

$$Sim_{wup}(c_1, c_2) = \frac{2 * N}{N_1 + N_2 + 2 * N}$$

where N_1 and N_2 indicate the distances that separate c_1 and c_2 (concepts) from the specific common concept, while N is the distance which separates the closest common ancestor of c_1 and c_2 from the root node (Slimani, 2013). The Wu-Palmer score has range $[0, 1]$. As reported by Slimani (2013), this metric is computationally simple yet it is as expressive as other theasurus based metrics.
To build the emoji description vectors we used the descriptions adopted by the Emojipedia; in two cases (U+1F383 and U+1F612), however, we replaced the primary description with a secondary description (obtained from the Unicode website) to ensure the retrieval of an existing match within Wordnet. Moreover, the words were stemmed and all the descriptions longer than one word were tokenized, and all the words not found in Wordnet were removed.
An example of a tweet vector (in italics) paired with an emoji description vector (in bold) is:
⟨*'come', 'celebrate', 'castle', 'get', 'special', 'offer'*⟩
⟨**'european', 'castle'**⟩
In this case the resulting distance vector is as follows:
⟨0.308, 0.235, *None*, *None*, 0.421, **1.0**, 0.105, 0.087, 0.167, 0.125, 0.1538 0.118⟩.
We can see that there one perfect correspondence (in bold) indicating an exact match between a word in the emoji description vector and a word in the tweet vector. The *None* values indicate that there is no existing path connecting the two words in WordNet.
Once the distances between the words in the tweet vector and the words in the emoji vector were computed, the highest value was kept as the resulting feature for the classification experiments. The assumption is that a maximum value close to 1 (ideally between 0.9 and 1) will indicate that the tweet vector contains a synonym of at least one word present in the emoji description.
Although thesaurus metrics have drawbacks, since they may establish high scores also among antonyms, this feature is expected to be effective in discriminating the Redundant class from the other classes.

Tf-Idf Bag of Words
A tf-idf matrix computed on unigram vectors represents each word in a document (a tweet in our case) as a weight which indicates how important that word is to identify the document's class. We adopted a tf-idf matrix computed on words combined with their POS tags. This choice has an impact on the number of features, which increases, and may include useful ones. However, as a trade off, the matrix will be sparser. This model does not consider word order and context, however we wanted to verify if a unigram model could improve over the baseline and potentially be used in combination with more complex features.

3.4. Classification

To perform the automatic classification we used the Python sklearn implementation of a Linear Support Vector Classifier (Pedregosa et al., 2011), set up for a multiclass classification.
To evaluate the proposed methodology we experimented with three different feature combinations and established two baselines to compare the different performances of our models.

Baseline I
The corpus is unbalanced since most of the instances are labeled as "Non-Redundant", therefore the lower bound for the classifier performance is given by assigning to each instance the most frequent class.

Baseline II
To verify if a unigram model can improve over a majority class baseline the classifier was trained on the tf-idf matrix built over the corpus unigrams combined with their POS tags.

SimPos
The third model is based on a combination of similarity and position features. This feature combination was expected to improve over both baselines.

Tf-Idf + SimPos
In this model the classifier was trained on the tf-idf matrix combined with SimPos.

LSA + SimPos
Lastly we performed dimensionality reduction by means of LSA on the tf-idf matrix and replicated the experiment on a 100-dimensional matrix combined with SimPos.

4. Results and discussion

Table 1 reports accuracy and F1 scores obtained with the same classifier on each of the illustrated models.

Model	Accuracy	F1 Score
Baseline I	59%	0.44
Baseline II	69.6%	0.66
SimPos	67.2%	0.64
Tf-idf+SimPos	71.8%	0.69
LSA+SimPos	**73%**	**0.7**

Table 1: Classification Results

The experiment was performed by means of a simple train test split where 50% of the corpus was used to train the classifier and 50% for testing. This approach was adopted following Zanzotto et al. (2011).

Comparison	McNemar's X^2
Baseline I – Baseline II	52.393
Baseline II – SimPos	86.218
Baseline I – SimPos	47.422
SimPos – Tf-idf+SimPos	88.347
Tf-idf+SimPos – LSA+SimPos	112.79

Table 2: Results of significance testing obtained using McNemar's X^2 test. In all cases, df=1, and $p < 0.001$.

The differences between the various models were tested pairwise by means of McNemar's X^2 test (Dietterich, 1998; Bostanci and Bostanci, 2013) and found statistically significant. The results of these tests are shown in Table 2.

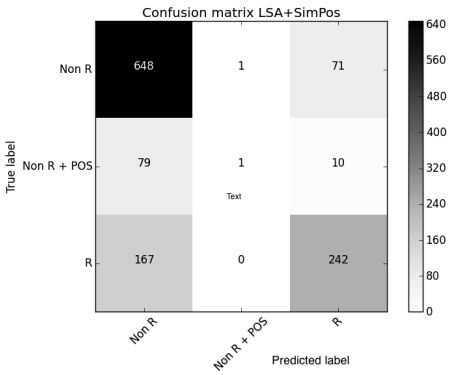

Figure 1: Confusion matrix showing the classifier's choices for the best performing model

In general, none of the models is good at recognizing instances of the least represented class (Non-Redundant + POS) for which only a minimal fraction of relevant instances is correctly labeled, suggesting that more training data are necessary to identify this class. Figure 1 shows the confusion matrix derived from the results of the best performing model, LSA+SimPos. If these results are compared with those obtained with the simpler models, it is clear that adding the semantic similarity measure increases the number of the instances of the Redundant class that are correctly classified (from 198 in the simplest model to 242). The remaining errors may be due to the fact that the emoji description vectors did not include the full Unicode emoji description, that distances were computed only among unigrams rather than n-grams. Furthermore, possible synonyms may be enclosed in a hashtag, thus it would be impossible to compute the similarity between hashtags and the emoji description vector without additional preprocessing, since hashtags may consist in a combination of words plus the hash symbol attached at the front (e.g. "#autumninjapan").

5. Conclusion

In the present work we investigated the informative behaviour of emoji in Twitter. The main interest was, in particular, in testing whether both human annotators and an automatic classifier can be trained to distinguish the use of emoji as being either redundant, non-redundant or as a word (thus, with syntactic properties). Furthermore, we were interested in determining whether specific features, such as the emoji position and the description of their content, could help a classifier in discriminating between these different conditions. We found the task to be difficult for human annotators. However, the results of our classification experiments show that the annotated corpus is still reliable enough for us to be able to obtain acceptable results. As regards the automatic classification, it emerged that the combination of the engineered features - SimPos - is more effective than the proposed majority class baseline in discriminating among the three classes. However, the best classification results were achieved when combining these features with tf-idf weights of the unigrams combined with their POS tags, and by applying dimensionality reduction to the resulting values.

To sum up, we have described a corpus of annotated tweets that can be used to study emoji usage with respect to whether they are redundant or add content to the text. Then we have demonstrated that automatic classification of emoji in tweets with respect to their redundancy can be achieved with an F-score of 0.7. Furthermore, we have shown that a combination of the unigrams tf-idf matrix reduced by means of LSA with position and similarity features is effective in reaching these results and performs better than two different baselines.

There are several aspects discussed in this work that may constitute a limitation and are, therefore, open to improvements and changes. First of all, the corpus is not very large even if it compares with the size of the dataset used in Zanzotto et al. (2011).To overcome this limitation the optimal solution would be to collect more data, both in terms of better coverage of the three classes and of different emoji tokens.

In the machine learning experiments we used 50% of the data for the training process and 50% for the test, following the setup proposed by Zanzotto et al. (2011). However, we are aware that for datasets the size of the presented one, a better approach would be to evaluate the model by means of cross validation. This will certainly be done in future.

More advanced models should also be tested, in particular models involving word embeddings. Alternatively, comparing similarity features obtained with different metrics (e.g. cosine similarity) and classification results obtained

by using different classifiers could also be a feasible approach to improve this research.

Further analysis can be done to verify whether the emoji that are mostly used in a redundant or non-redundant way belong to a specific semantic category. Furthermore, it could be interesting to test a simplified design, by collapsing the Non-Redundant and the Non-Redundant + POS in a single category, thus implementing a binary classification only concerned with the two broad semantic categories.

6. Acknowledgements

We would like to thank George Giannakopoulos, the Institute of Informatics and Telecommunications at NSCR Demokritos in Athens and everybody at Sci.FY.

7. Bibliographical References

Barbieri, F., Ronzano, F., and Saggion, H. (2016). What does this emoji mean? a vector space skip-gram model for twitter emojis. In *Language Resources and Evaluation conference, LREC*, Portoroz, Slovenia, May.

Boia, M., Faltings, B., Musat, C.-C., and Pu, P. (2013). A:) is worth a thousand words: How people attach sentiment to emoticons and words in tweets. In *Social Computing (SocialCom), 2013 International Conference on*, pages 345–350. IEEE.

Bostanci, B. and Bostanci, E. (2013). An evaluation of classification algorithms using Mc Nemar's test. In *Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012)*, pages 15–26. Springer India.

Derczynski, L., Ritter, A., Clark, S., and Bontcheva, K. (2013). Twitter part-of-speech tagging for all: Overcoming sparse and noisy data. In *RANLP*, pages 198–206.

Derks, D., Bos, A. E., and Von Grumbkow, J. (2007). Emoticons and online message interpretation. *Social Science Computer Review*.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923, October.

Donato, G. and Paggio, P. (2017). Investigating redundancy in emoji use: Study on a twitter based corpus. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 118–126, Copenhagen, Denmark, September. Association for Computational Linguistics.

Eisner, B., Rocktäschel, T., Augenstein, I., Bošnjak, M., and Riedel, S. (2016). emoji2vec: Learning emoji representations from their description. *arXiv preprint arXiv:1609.08359*.

Hallsmar, F. and Palm, J. (2016). Multi-class sentiment classification on twitter using an emoji training heuristic.

Kelly, R. and Watts, L. (2015). Characterising the inventive appropriation of emoji as relationally meaningful in mediated close personal relationships. *Experiences of Technology Appropriation: Unanticipated Users, Usage, Circumstances, and Design*.

Kelly, C. (2015). Do you know what i mean >:(: A linguistic study of the understanding of emoticons and emojis in text messages.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Miller, H., Thebault-Spieker, J., Chang, S., Johnson, I., Terveen, L., and Hecht, B. (2016). "blissfully happy" or "ready to fight": Varying interpretations of emoji. *ICWSM'16*.

Novak, P. K., Smailović, J., Sluban, B., and Mozetič, I. (2015). Sentiment of emojis. *PloS one*, 10(12):e0144296.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830.

Slimani, T. (2013). Description and evaluation of semantic similarity measures approaches. *arXiv preprint arXiv:1310.8059*.

Swiftkey. (2015). Swiftkey emoji report april 2015. https://blog.swiftkey.com/americans-love-skulls-brazilians-love-cats-swiftkey-emoji-meanings-report/.

Zanzotto, F. M., Pennacchiotti, M., and Tsioutsiouliklis, K. (2011). Linguistic redundancy in twitter. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 659–669. Association for Computational Linguistics.