What does this Emoji Mean?
A Vector Space Skip-Gram Model for Twitter Emojis

Francesco Barbieri, Francesco Ronzano, Horacio Saggion
Universitat Pompeu Fabra, Barcelona, Spain
Tractament Automàtic del Llenguatge Natural (TALN)
name.surname@upf.edu

Abstract

Emojis allow us to describe objects, situations and even feelings with small images, providing a visual and quick way to communicate. In this paper, we analyse emojis used in Twitter with distributional semantic models. We retrieve 10 millions tweets posted by USA users, and we build several skip gram word embedding models by mapping in the same vectorial space both words and emojis. We test our models with semantic similarity experiments, comparing the output of our models with human assessment. We also carry out an exhaustive qualitative evaluation, showing interesting results.

Keywords: Emojis, Social Networks, Embeddings

1. Introduction

Writing and reading short texts is part of our daily life. There are very popular platforms on which we rely to communicate our interests, opinions, emotions and daily activities using short texts, including Twitter, Instagram and WhatsApp. The users of such platforms are extremely diverse with respect to their writing style and, more in general, with respect to the way they communicate with each other on these Social Media services. Yet, there is an aspect they share: the use of emojis. Emojis[1] are ideograms and smiles that can be considered the natural evolution of the emoticons (like `:)` and `:D`). There are several types of emojis, ranging from facial expressions to animals, objects or places. Emojis use is increasing every day, rapidly changing the way in which we communicate in Social Networks. Emojis are used to communicate simple things or feelings in a fresh, visual and condensed way.

There are few studies that perform in depth analyses of emojis.[2]. We study emojis in the context of Twitter. Three tweets that include Twitter emojis[3] are shown in Figure 1.

Emojis can be used instead of words, like in the tweet of Kate Perry, where the emoji 🇺🇸 is used instead of *America*. Emojis are also exploited to denote the sentiment and real intents of a message, like in the Dwayne Johnson's tweet. In his tweet, he makes a joke, suggesting to eat pancakes as they make you strong (denoting this with the arm emoji 💪), but at the end of the tweet he laughs (using 😂), suggesting that he is joking.

In the tweet of Dawayne Johnson we can also see another emoji, a brown square 🟫. In this context it is used to symbolize a rock (as his stage name is *The Rock*), and close to the arm emoji, it would probably means "strong like a rock", like himself. The use of emojis strongly depends on both the context and the community to which the same emoji is addressed. Even if the Unicode Consortium[4] decides the official meanings of the emojis, the use of emojis online is unpredictable and socially-determined. Emoji meanings change rapidly over time and can not be defined statically.

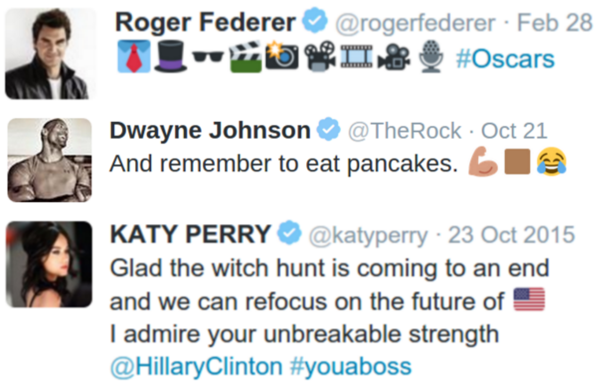

Figure 1: Examples of tweets that include emojis.

The tweet of Roger Federer (see Figure 1) is a clear example of evolution of language in Twitter. The tweet do not even include words (apart from #Oscars), yet it still communicates several things. It says that the tennis player will assist the Oscars ceremony, he will dress elegant suit 👔🎩, and there will be photos 📷 (red carpet) and movies 🎥.

In this paper, we generate, validate and share semantic vectorial models that are built over tweets by consistently mapping in the same vectorial space both words and emojis. In particular, we propose distinct approaches to jointly generates vectorial embeddings of both words and emojis of tweets. We validate the effectiveness of our models (in embedding the semantics of emojis) by comparing similarity scores among words and emojis defined by humans with the same scores derived by our model. The models are available at `http://sempub.taln.upf.edu/tw/emojis/`

[1] Emojis comes originally from Japanese web texts.

[2] While very good analysis has been carried out in the context of industries `http://instagram-engineering.tumblr.com/post/117889701472/emojineering-part-1-machine-learning-for-emoji`

[3] `https://blog.twitter.com/2014/open-sourcing-twitter-emoji-for-everyone`

[4] `http://unicode.org/`

2. Related work

Currently, emojis represent a widespread and pervasive global communication device largely adopted by almost any Social Media service and instant messaging platform (Jibril and Abdullah, 2013; Park et al., 2013; Park et al., 2014). Emojis (like the older emoticons) support the possibility to express diverse types of contents in a visual, concise and appealing way that is perfectly suited to the informal style of social media communications. The meaning expressed by emoticons has been exploited to enable or improve several tasks related to the automated analysis of Social Media contents, like sentiment analysis (Hogenboom et al., 2015; Hogenboom et al., 2013) or irony detection (Reyes et al., 2013; Barbieri et al., 2014). In this context, emoticons have also been often exploited to label and thus characterize the textual excerpts where they occur. As a consequence, by analyzing all the textual contents where a specific emoticon appears several language resources have been built. In this context, Yang et al. (2007) propose a method to create an emotional lexicons by relying on the textual contents occurring together with emoticons in messages published in the Yahoo! Kimo Blog Service. Tang et al. (2014) build a sentiment lexicon customized to Twitter by training a neural network thanks to tweets labeled with positive and negative emoticons. Boia et al. (2013) analyze sentiment lexicons generated by considering emoticons showing that in many cases they do not outperform lexicons created only with textual features. Go et al. (2009) and Castellucci et al. (2015) use distant supervision over emotion-labeled textual contents in order to respectively train a sentiment classifier and build a polarity lexicon.
In order to better compare the meaning of emoticons, recently new approaches to model the information they convey have been proposed. Aoki and Uchida (2011) describe a methodology to represent each emoticon as a vector of emotions. Each element of the vector of an emoticon corresponds to the relevance of a specific emotion to characterize that emoticon. Emoticon vectors have been populated by analyzing the emotional words occurring together with each emoticon. Jiang et al. (2015) propose an approach that relies on word2vect (Mikolov et al., 2013b) to build a distributional semantic vectorial space where to represent and compare emojis. Cappallo et al. (2015) proposed Image2Emoji, a multimodal approach for generating emoji labels for images.

3. Dataset and Text Analysis

To support the creation of the semantic vectorial models presented in this paper we gathered a dataset composed of more than ten millions tweets retrieved with the Twitter APIs [5]. We retrieved geo located tweets that were posted from United States of America, between October 2015 and February 2016. We decided to only use geo located tweets in order to retrieve tweets from real user, filtering out spam and bot generated tweets.
In order to preprocess the text we employed the `CMU Tweet Twokenizer`[6](Gimpel et al., 2011). We also incorporated a filtering step in which we removed stopwords, punctuation marks (but leaving emoticons like `:)` or `:P`), Twitter hashtags and user mentions as they were not relevant to the task. We also lowercased all the tweets to reduce noise.

[5] `https://dev.twitter.com`
[6] `http://www.ark.cs.cmu.edu/TweetNLP/`

Filter	Tweets	Tokens Avg
raw	9,862,837	15.3
clean	9,514,951	7.5
onlyemo	573,007	3.3

Table 1: Number of tweets and average number of tokens per tweet in the dataset filtered with the three methods.

4. Vector Space Models

We employed the skip-gram neural embedding model introduced by Mikolov et al. (Mikolov et al., 2013a).
We built several variants of the skip-gram model by training with different parameters (dimensions of the vectors and length of the window) to find the best configuration for our task. The dimension of the vectors that better fitted the task was 300. We tested several models with dimensions between 50 and 700, and 300 turned out to be the one that leads to better results in our evaluation experiments (see Section 5.). Regarding the window size, we experimented with window lengths varying from 3 to 12 tokens. We also wanted to explore weather the semantics of emojis is affected by all the elements of its context (punctuation and words), only by the words, or only by other emojis co-occurring in the same tweet. For this reason we trained our models on the datasets cleaned with three filters. The **raw** filter removes links and mentions, leaving all the rest of the context of the tweets, emoticons and punctuation included. The **clean** filter also removes all the punctuation and stop words of the tweet. The **only emojis** filter leaves in the dataset only emojis. In Table 1 we report the number of tweets and average tokens for the dataset filtered with the three methods. Applying the raw filter we obtain a dataset of vocabulary 600,141 tokens. The vocabulary of the dataset filtered with the clean filter includes 187,308 tokens, while if filtered with only emojis 856 tokens (the emojis that our model includes).

5. Experiments and Evaluation

We performed two different experiments to evaluate our system. In the first experiment we carried out a quantitative evaluation and in the second experiment a qualitative evaluation.

5.1. Quantitative Evaluation

We performed a pair similarity and relatedness test, a common practice in embeddings evaluations (Mikolov et al., 2013a; Levy and Goldberg, 2014; Baroni et al., 2014). While there are currently many shared and widely used dataset that model word similarity (Rubenstein and Goodenough, 1965; Miller and Charles, 1991; Finkelstein et al., 2001), in our knowledge, there are not previous similarity tests of Emojis. We compiled EmoTwi50, a dataset that

Filter	Window	Sim	Rel	Avg
raw	3	0.733	**0.788**	0.796
raw	6	0.749	0.787	0.804
raw	9	0.757	0.787	0.808
raw	12	0.738	0.775	0.792
clean	3	**0.777**	0.780	**0.815**
clean	6	0.772	0.785	**0.815**
clean	9	0.763	0.770	0.803
clean	12	0.766	0.772	0.805
onlyemo	3	0.618	0.666	0.672
onlyemo	6	0.628	0.678	0.683
onlyemo	9	0.616	0.663	0.669
onlyemo	12	0.630	0.672	0.681

Table 2: Quantitative evaluation: Similarity, Relatedness and Average Pearson Correlation between human gold standard and five different vector space models. The dimensions of the vectors is 300.

contains a set of 50 pairs of emojis with degrees of *similarity* (functional similarity) and *relatedness* (topical similarity) (Levy and Goldberg, 2014). In order to have human evaluated scores, we performed an experiment with 8 participants. Each participant was asked to rate each pair by assigning a score of similarity and a score of relatedness, both ranging from 1 to 4. In particular, for each pair, the participant was asked if he/she agreed with the following statements (1 = totally disagree, 4 = totally agree):

- I consider the two emojis equivalent (similarity)
- I can imagine a situation in which I would use the two emojis together (relatedness)

All the participants were familiar with Twitter language and the use of emojis. Two of the participants were female and six male, and the age was distributed between 25 and 52 years old.
To compile the list of pairs to evaluate, we selected the 100 most frequent emojis in our dataset. At the moment of pairing the emojis we had to deal with one issue: choosing the pairs randomly leads to an unbalanced distribution, since most of the emojis are not similar to each other (we empirically measured that, in average, only one out of 20 emojis is very similar). Hence, we selected 25 pairs randomly and 25 pairs were chosen by one of the authors, selecting similar emojis (in order to balance the distribution). The human intervention was not an issue as the pairs were later validated by other people[7]. The correlation between the participants was strong, the average pairwise Pearson correlation was 0.71 (0.76 in the similarity task and 0.66 in relatedness). The gold standard of each pair was given by the average of the scores of each participant.
In Table 2 are shown the Pearson correlations between the human gold standard and the similarity scores given by our models (for each pair we measure the cosine similarity of the vectors of the two emojis). Three types of correlations are shown, Similarity, Relatedness and Average between the previous two.
We tested our model combining window sizes from 3 to 12, and the type of filter of the dataset.
Looking at the average correlation, we can see that models with cleaned words are better than the models trained only on emojis or on the raw dataset.
In spite of this, the only-emojis models showed very good correlation considering that there were no other information to characterize the context of each emoji. These good results are probably coming from the common practice of using more than one emoji at the same time (similar or related). We also have to say that the only emojis models had less tweets for training, as this dataset included only the tweets with at least two emojis (see Table 1)
The average results of the raw dataset are worse but comparable with the clean dataset. The raw dataset performs well considering the amount of noise that includes. Moreover, the best relatedness scores are achieved with the raw dataset (even if very close to the clean results 0.788 versus 0.785). Vectos build on the raw dataset can effectively model the relatedness of emojis but are less effective in modelling their similarity.
On the other hand the clean dataset seems more balanced and with good results. Hence, the best way to filter the dataset is to remove punctuation and stop words, thus exploiting only words and other emojis to characterize the context of each emoji.
Since tweets are very short texts, we expected that a small context window would work better. Indeed in the Table 2, the best window sizes are 3 and 6. We also tested on bigger windows, but none of the model learned better than with small windows. This may be related also to the fact that the average length of the tweets is comparable to the better performing window sizes.
Where the types of correlations are concerned, we observe that our models are better correlated to Relatedness scores than Similarity scores, suggesting that our models learn context relations better than similarity ones. This was expected as the relatedness task (*I would use the two emojis together*) is easier as it only asks to measure if the emojis are often used together. The similarity task is harder, since it demands an abstraction over the meaning of the emojis, asking if the emojis are semantically similar, i.e. used in similar context even if not together.
However, all the models are closest to the average scores, suggesting that the models learn a sort of combination of Similarity and Relatedness. This could happen because if we find two, or more, emojis in a tweets, sometimes they are the same or similar emojis used to stress a concept; other times they are different emojis used in combination to formulate a more complex concept. In both cases these pairs of emojis are considered by the word2vec algorithm as belonging each on to the context of the other and thus they reciprocally influence the creation of both embedding vectors.

5.2. Qualitative Evaluation

This evaluation was performed to see the quality of the vectors that represent each emoji. This qualitative evaluation is

[7]The person who chose the 25 non-random pairs did not participate to the experiment.

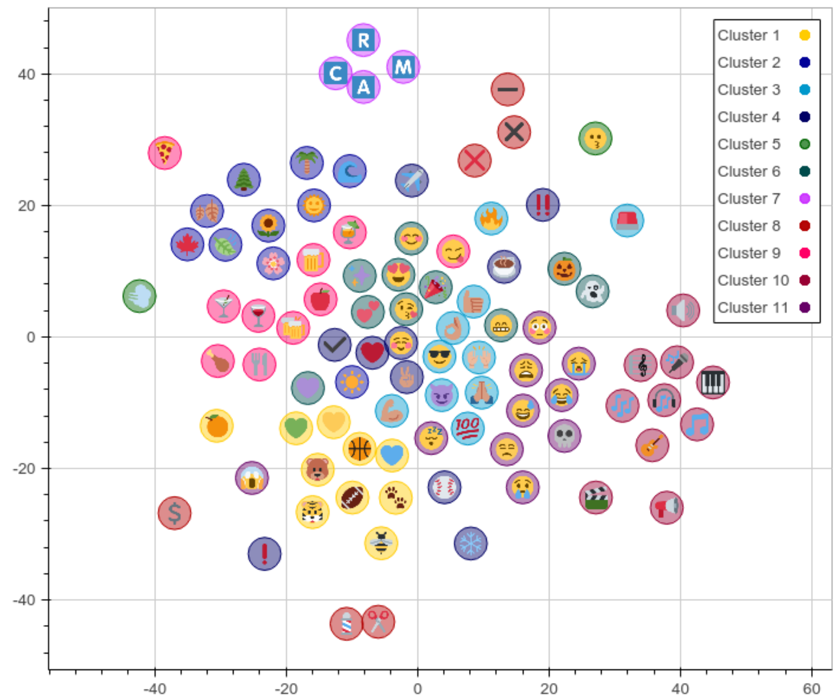

Figure 2: Visualisation of 100 emojis vectors, reduced to two dimensions with t-SNE. Similar emojis get clustered together. The model used is the clean filter with window size 6.

Emoji	Most similar text tokens
😘	love, babe, youu, awww, bby
😋	mmm, craving, pancakes, yummmmm, nutella
😳	dying, wtf, nooo, noo, lmaooo
😭	mlrt, omfg, blum, lmaoooo, crying
😢	ugh, sad, stomach, miss, noooooo
☕	coffee, roasters, caffeine, latte, redeye
⚽	soccer, futbol, regionals, mckale, muskingum
🗽	statue, nyc, rockerfeller, barneys, hk
❄️	snowing, snow, icy, brrrrr, outerwear
👭	sistas, ily, bunches, sista, ilysm

Table 3: The five text tokens that better characterize each cluster shown in Figure 2. We used the best model (clean with window 6)

composed of two parts. In the first part we take in consideration the emojis singularly, while in the second part we cluster the emojis and study groups of related emojis.

5.2.1. Single emojis

This subsection presents two experiments. Firstly we plot the emojis and look if similar emojis are plotted close to each other, and secondly we look at the relation between words and emojis.

In the first part of the qualitative evaluation the vectors of the 100 emojis were reduced to two dimensions and plotted in the same space. We used the t-Distributed Stochastic Neighbor Embedding (t-SNE), a technique to reduce dimensionality of high-dimensional datasets (Van der Maaten and Hinton, 2008). In Figure 2 we can see that reducing 300 dimensions to only two introduce a bit of noise and some emojis are out of context (for example the pizza is not close to the other food related emojis). Overall the vectors seem of a reasonable quality in grouping together similar emojis. Most of the yellow face emojis are together but correctly separated from sad faces (as crying and indignant faces) and happy faces (link blinks and smiles). Kissing yellow faces are close to love related emojis (hearts, friends).

In the second part of the Qualitative evaluation we used the best model with words (clean filter with window size 6) to evaluate association between text tokens and emojis. We selected five facial expressions and five objects, people and places emojis. For each example we show the five most similar text tokens (Table 3). We can see that most of the facial expressions are not always associated to words that describe their meaning (with exception of “crying” in 😭). However, related terms can easily be associated to the meaning conveyed by emojis, like “mmm” and “yummmmmm” for 😋 and “love” and “babe” for 😘. In the other five examples, there are not only good related tokens, like “nyc”, “brrrrr” and “ily” (i love you), but also good similar token that describe the emoji, such as “coffee” for ☕, “statue” for 🗽 and “snow” for ❄️.

5.2.2. Clusters of emojis

We explore in this section cluster of emojis, to see whether we can group emojis by topics. Our hypothesis is that if the emoji vectors are well constructed we should be able to group similar emojis together. We also have to consider that some emojis can be used for different purposes and in different context (just like words). For instance, the emoji 🙏 is used to ask something (“please”), to indicate the action

Cluster	Topic	Closest words
1	Sports and animals	szn, dowdy, modis, lambeau, homecoming
2	Nature	fairleigh, latepost, ilysm, botanic, tau
3	Body gestures and positive	take1lounge, quavo, powered, tatted, gentlemens
4	Free time	alief, arif, sneakers, tonight, med
5	Unclear	carpet, i80, bussers, bergstrom, trw
6	Love and parties	ilysm, love, rho, alpha, ytd
7	Letters	meatpacking, makefield, cucamonga, mindcare, esta
8	Barber and simbols	attention, cutz, latepost, mcd2, tonight
9	Eating and drinking	amore, mongolianbbq, apple, pineapple, momofuku
10	Music	edc, gentlemans, song, jumpman, hobojoesjuice
11	Sad and tears	sleep, tired, bipolar, feelings, hurt

Table 5: The five text tokens that better characterize each cluster shown in Figure 2

Cluster	Elements	Percentage
1	20	6.6
2	28	9.3
3	53	17.6
4	28	9.3
5	2	0.6
6	88	29.3
7	4	1.3
8	6	2.0
9	30	10.0
10	10	3.3
11	31	10.3

Table 4: Number of elements of each cluster and percentage over the whole corpus.

of praying or to high five [8].
We build 11 clusters[9] with K-means from the 300 most frequent emojis. The size of the clusters are shown in Table 4. The biggest cluster is composed of 88 emojis (about 30% of the 300 emojis) and the smallest includes two emojis. In the t-SNE plot (Figure 2) we added the color of the cluster in the back of each emoji. The emojis plotted are the 10 emojis closest to the centroid of each cluster.
At first glance, we can see that the clusters seem to have a specific identity, like the music related emojis (Cluster 10) and the sad yellow faces (Cluster 11). In the Topic column of Table 5 we reported possible labels for each clusters (the labels were decided by the authors). The worst clusters in terms of similarity between members, seem to be the number 5, which includes only 2 emojis that do not seem related and number 8, which includes barber emojis and symbols like crosses and horizontal lines. The rest of the clusters have a clear identity and seems to be quite consistent. The fact that we obtain a highly populated cluster probably indicates the need to exploit hierarchical clustering approaches to further explore this aspect of emojis embeddings.
In Table 5 we also report the words related to each clusters. We tried different methods to explore the relations between clusters and words. We firstly tried to select the most frequent words among the N (tested with N = 3 to 100) closest words to each emojis. The resulting words of this method were noisy, probably because clusters encode more than one topic, and frequency is not a good filter. We also tried to simply take the N closest words to each centroid of the cluster, but it was noisy as well, as the words selected were not necessary the closest words to the members of the cluster. We finally found the best results (that are still slightly noisy) by combining the two methods. We select for each cluster the 10 closest emojis to the centroid, take for each of these emojis the 30 closest words, and then select the 10 most frequent words of this set of 300 words.

The related words to the clusters seem consistent even if there is noise. In cluster 1 words *szn* (season) and *lambeau* are both related to sports, *botanic* correctly describe the Nature cluster, and *powered* is a good example for the body gestures (arm emoji for instance). Words related to clusters 4, 5, 7 and 8 are the most noisy, while for the other clusters the words are somehow related to the cluster topic (e.g. *ilysm* for Love cluster, *apple* and other food for Eating and Drinking). The most clear results are the words of Cluster 11, that are all related to bad feelings or being tired. Finally two peculiar result are the words *amore* for the food cluster (the italian word for love is apparently related to food in USA) and *bipolar* in Cluster 11. The latter seems consistent as the Cluster 11 includes laughing 😂 and crying 😭 emojis (both with tears, for this reason in the same cluster).

6. Conclusions

In this paper we studied Twitter emojis with embedding models. To the best of our knowledge no previous study has concentrated on the analysis of Twitter to model emojis semantics. We make the vectorial models of both words and emojis available online to the research community. As future work we plan to look for different and more sophisticated algorithms to associate words and emojis by exploring additional information, for example in lexical resources. Moreover, it would be interesting to study the sociolinguistic aspects of the use of emojis. For instance the use of emojis of users who belong to different communities and cultures, or how the meaning of an emoji evolves.

[8] http://uk.businessinsider.com/the-prayer-hands-emoji-is-changing-2015-3?r=US&IR=T

[9] We tried different number of clusters, and 11 seemed to create the least noisy clusters.

Acknowledgments

We thank the three anonymous reviewers for the valuable and interesting suggestions, specially for the future work. We also thank the annotators Alicia, Hamdi, Juan, Luis, Maria and Simon.

This work is (partly) supported by the Spanish Ministry of Economy and Competitiveness under the Maria de Maeztu Units of Excellence Programme (MDM-2015-0502). The second and third author acknowledge support from the EU project Dr. Inventor (FP7-ICT-2013.8.1 project number 611383).

7. Bibliographical References

Aoki, S. and Uchida, O. (2011). A method for automatically generating the emotional vectors of emoticons using weblog articles. In *Proc. 10th WSEAS Int. Conf. on Applied Computer and Applied Computational Science, Stevens Point, Wisconsin, USA*, pages 132–136.

Barbieri, F., Saggion, H., and Ronzano, F. (2014). Modelling sarcasm in twitter, a novel approach. *ACL 2014*, page 50.

Baroni, M., Dinu, G., and Kruszewski, G. (2014). Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 238–247.

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.

Cappallo, S., Mensink, T., and Snoek, C. G. (2015). Image2emoji: Zero-shot emoji prediction for visual media. In *Proceedings of the 23rd Annual ACM Conference on Multimedia Conference*, pages 1311–1314. ACM.

Castellucci, G., Croce, D., and Basili, R. (2015). Acquiring a large scale polarity lexicon through unsupervised distributional methods. In *Natural Language Processing and Information Systems*, pages 73–86. Springer.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414. ACM.

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47. Association for Computational Linguistics.

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

Hogenboom, A., Bal, D., Frasincar, F., Bal, M., de Jong, F., and Kaymak, U. (2013). Exploiting emoticons in sentiment analysis. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, pages 703–710. ACM.

Hogenboom, A., Bal, D., Frasincar, F., Bal, M., De Jong, F., and Kaymak, U. (2015). Exploiting emoticons in polarity classification of text. *J. Web Eng.*, 14(1&2):22–40.

Jiang, F., Liu, Y.-Q., Luan, H.-B., Sun, J.-S., Zhu, X., Zhang, M., and Ma, S.-P. (2015). Microblog sentiment analysis with emoticon space model. *Journal of Computer Science and Technology*, 30(5):1120–1129.

Jibril, T. A. and Abdullah, M. H. (2013). Relevance of emoticons in computer-mediated communication contexts: An overview. *Asian Social Science*, 9(4):201.

Levy, O. and Goldberg, Y. (2014). Dependencybased word embeddings.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Miller, G. A. and Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28.

Park, J., Barash, V., Fink, C., and Cha, M. (2013). Emoticon style: Interpreting differences in emoticons across cultures. In *ICWSM*.

Park, J., Baek, Y. M., and Cha, M. (2014). Cross-cultural comparison of nonverbal cues in emoticons on twitter: Evidence from big data analysis. *Journal of Communication*, 64(2):333–354.

Reyes, A., Rosso, P., and Veale, T. (2013). A multidimensional approach for detecting irony in twitter. *Language resources and evaluation*, 47(1):239–268.

Rubenstein, H. and Goodenough, J. B. (1965). Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Tang, D., Wei, F., Qin, B., Zhou, M., and Liu, T. (2014). Building large-scale twitter-specific sentiment lexicon: A representation learning approach. In *COLING*, pages 172–182.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85.

Yang, C., Lin, K. H.-Y., and Chen, H.-H. (2007). Building emotion lexicon from weblog corpora. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 133–136. Association for Computational Linguistics.