Exploiting Emojis for Sarcasm Detection

4 authors:

Jayashree Subramanian
Arizona State University

Varun Sridharan
Arizona State University

Kai Shu
Illinois Institute of Technology

Huan Liu
Arizona State University

Exploiting Emojis for Sarcasm Detection

Jayashree Subramanian*, Varun Sridharan*, Kai Shu and Huan Liu

Computer Science and Engineering, Arizona State University, Tempe, AZ, USA
{jsubram5, vsridh19, kai.shu, huan.liu}@asu.edu

Abstract. Modern social media platforms largely rely on text. However, the written text lacks the emotional cues of spoken and face-to-face dialogue, ambiguities are common, which is exacerbated in the short, informal nature of many social media posts. Sarcasm represents the nuanced form of language that individuals state the opposite of what is implied. Sarcasm detection on social media is important for users to understand the underlying messages. The majority of existing sarcasm detection algorithms focus on text information; while emotion information expressed such as emojis are ignored. In real scenarios, emojis are widely used as emotion signals, which have great potentials to advance sarcasm detection. Therefore, in this paper, we study the novel problem of exploiting emojis for sarcasm detection on social media. We propose a new framework ESD, which simultaneously captures various signals from text and emojis for sarcasm detection. Experimental results on real-world datasets demonstrate the effectiveness of the proposed framework.

1 Introduction

Social media plays a major role in everyday communication. While images and videos are common in social media sites such as Facebook [1] and Twitter [2], the text is still dominating the communication. Communication through text may lack non-verbal cues, and *emojis* can provide richer expression to mitigate this issue. Emojis are a set of reserved characters that are rendered as small pictograms that depict a facial expression [1, 11]. In social media, sarcasm represents the nuanced form of language that individuals state the opposite of what is implied.

Sarcasm detection is an important task to improve the quality of online communication. First, it helps us to understand the real intention of the user's feedback. For example, user reviews can contain examples such as 'Wow this product is great", "It is very fast", "Totally worth it", etc. These comments, however, are being said in a sarcastic tone. Second, sarcastic posts may influence people's emotions and reactions to the political campaign [10].

The majority of existing sarcasm detection algorithms focuses on text information [8]. These include identifying the traits of the user from their past activities, responses texts, etc. Most of them have tried to train deep neural network models using the text to analyze sarcasm. To overcome the challenges faced by all of these methods and for better performance, the Emoji can be considered to detect sarcasm. Emojis help us to find the tone of speech, the mood

[1] https://www.facebook.com/
[2] https://twitter.com/?lang=en

of the user and identify sarcasm in a better way. For example, comments such as, “Wow!! This is beautiful 😜 😜” , “You can do this, I trust you 😆 😜”, “It’s big proud 😊” are examples of sarcastic comments. The above comments without the emoji convey us a different meaning and are taken in the positive sense since it has the keywords “beautiful”, “proud”, “trust”, “wow”. However, with emoji, they strongly help us to identify the sarcasm in the comments. Human thought process and emotions are best conveyed through Emojis and these emotional signals are much stronger than the text. These emotional signals will help the model to learn more accurately about the intention, thought process of the user than by merely looking at the text.

In this paper, we address the problem of identifying sarcasm in social media data by exploiting Emojis. In essence, we investigate: 1) how to learn the representation of text and emojis separately; 2) how to take advantage of the emoji signals to improve sarcasm detection performance. In an attempt to solve these two challenges, we propose a novel Emoji-based Sarcasm Detection framework ESD, which captures text and emoji signals simultaneously for sarcasm detection. Our contributions are summarized as follows:

– We provide a principled way to model emoji signals for social media post;
– We propose a new framework ESD which integrates text and emoji signals into a coherent model for sarcasm detection; and
– We conduct experiments on real-world datasets to demonstrate the effectiveness of the proposed framework ESD.

2 Related Work

2.1 Sarcasm Detection

Automatic sarcasm detection is the task of predicting sarcasm in text. It is an important step in sentiment analysis, considering the prevalence and challenges of sarcasm in a sentiment-bearing text [8]. [4] identifies sarcasm using bi-directional recurrent neural network by extracting all the contextual features from the history of tweets. [3] The sarcasm is handled by creating word embeddings for the tweets and is fed to the DNN, CNN, RNN, LSTM models. The performance of these models are compared with each other and it is observed that the combination of CNN + LSTM + DNN gives the highest F1 score. However, the built model fails to classify comments like ’Thank God it is Monday!’ as sarcastic comments. [6] This paper investigates how sentiment, emotional, personality features can be combined to detect sarcasm using deep Convolutional neural networks. The sarcasm is identified using the user’s past activities, behavioral and psychological features. The SCUBA model performs with an accuracy of 82 percent with all features when compared to other baseline models [14].

2.2 Emoji Analysis Of Social Media

Emojis have become an important tool that helps people to communicate and express their emotions. The study of emojis, as they pertain to sentiment classification and text understanding, attracts attention [1,9,12]. Hu *et al.* [7] proposes an unsupervised framework for sentiment classification by incorporating emoticon signals. Hallsmar *et al.* [5] investigates the feasibility of an emoji training heuristic

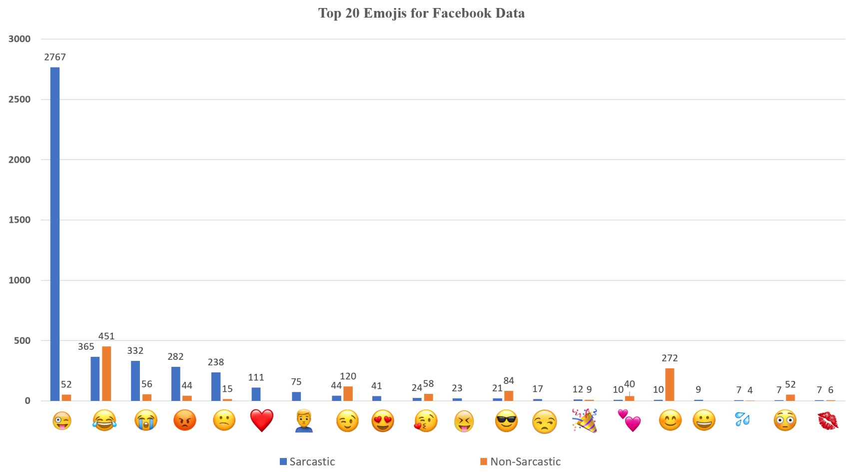

Fig. 1. Comparison of top 20 Emojis for Facebook data.

for multi-class sentiment analysis on Twitter with a Multinomial Naive Bayes Classifier. Eisner *et al.* [2] learns emoji representation by running skip gram on descriptions of emojis provided in the Unicode standard. The resulting emoji representation along with the word embedding from Google News are used to perform sentiment analysis and the results show that emoji representation can improve sentiment analysis. Kelly *et al.* [9] shows that emojis can be used as appropriations which can help facilitate communications using the interview data.

3 Preliminary Analysis Of Emoji Usage

Emojis serve as a medium for us to express certain opinions that can't be expressed by our voice or body language. Emojis are the major contributing factor to the improvement in accuracy of our model because the neural network learns the connection between text and emojis. This analysis is performed to research in depth about the types of emojis used across the comments in the Twitter and Facebook data set. This gives us a clear picture of the most frequently used emojis in both sarcastic as well as non-sarcastic comments which in turn helps us to rank emojis based on their count of occurrences in the comments. The top 20 and top 5 emojis used in our Twitter/Facebook data are visualized through the graphs. The following insights are obtained from the graphs.

- On comparison of emojis used across entire Facebook and Twitter data, the usage of Face with tongue out emoji is the highest (2.7K) among the sarcastic comments. The Face with tears of Joy, Loud crying face (2.6K), Grinning and Pouting face are the three specific emojis that are most frequently used with non-sarcastic comments.
- The number of other emojis used in sarcastic comments like winking face, the smirking face is found to be uniformly distributed across the Twitter data

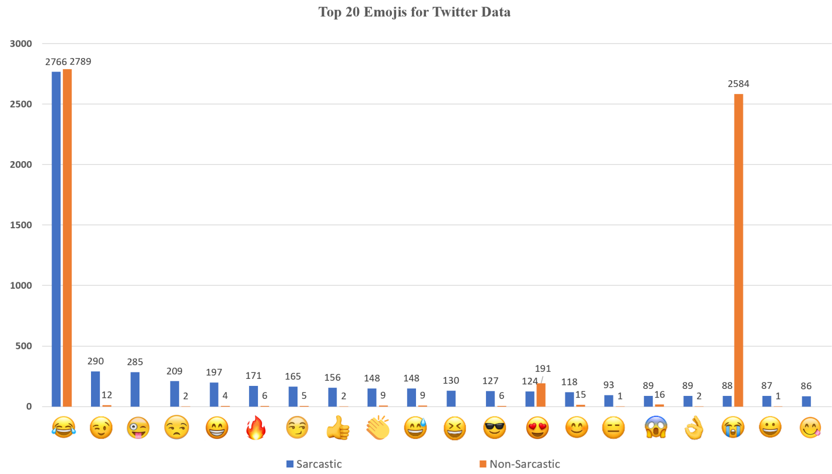

Fig. 2. Comparison of top 20 Emojis for Twitter data.

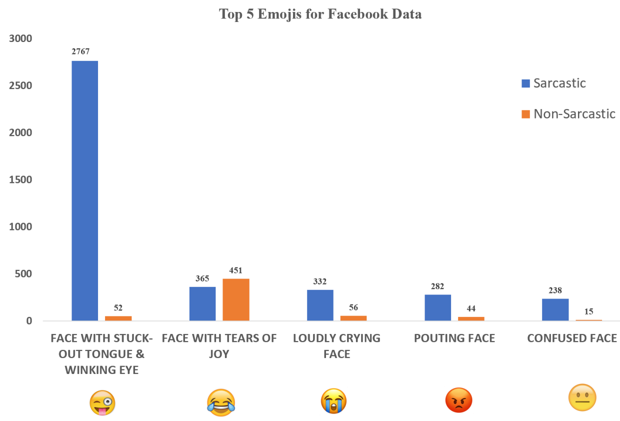

Fig. 3. Comparison of top 5 Emojis for Facebook data.

whereas emojis such as the loud crying face, pouting face, the confused face is observed to be uniformly distributed for the Facebook data.

– The usage of Face with stuck out tongue emoji is the first highest for Facebook data and third highest for Twitter data. However, the face with tears of joy emoji is being increasingly used in both sarcastic and non-sarcastic comments across the platforms.
– It is also clearly observed that the amount of Face with tongue out emoji in sarcastic comments is very high which is nearly 54 times its usage in non-sarcastic comments for Facebook data. For twitter non-sarcastic comments, the count of this emoji is in-fact zero. This proves the fact that most of the comments having this emoji are clearly being sarcastic in nature.

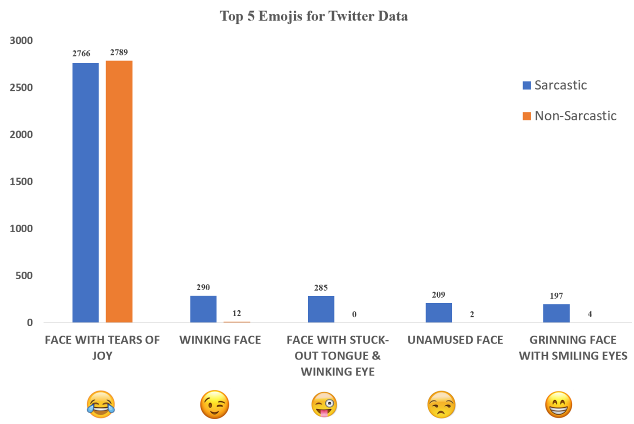

Fig. 4. Comparison of top 5 Emojis for Twitter data.

4 Text and Emoji Embedding for Sarcasm Detection

In this section, we introduce the details of the proposed framework ESD for sarcasm detection on social media. It mainly consists of three components (see Figure 2): a text encoder, an emoji encoder, and a sarcasm prediction component. In general, the text encoder describes the mapping of words to latent representations; the emoji encoder illustrates the extraction of emoji latent representations, and the sarcasm prediction component learns a classification function to predict sarcasm in social media posts.

4.1 Text Encoder

The entire dataset, which is basically a list of sentence vectors, is divided into a list of tokens which contains both English words as well as Emojis. In order to get the embedding for the English words alone, we first filter the data with the help of python libraries like Enchant and the NLTK work tokenizer.

The embeddings are obtained from GloVe [13] and are stored in an embedding matrix W. Essentially, an embedding is a mapping from a word to a vector. Therefore, it is important to store the embeddings in the same order that they occur in the sentence. These embeddings represent the words in a transformed space. This helps our model to capture relationships between words which are not possible otherwise. Given a list of sentence vectors S_i, where each sentence contains T_i words, there exists a word embedding $\tilde{w_i}$ for every word w_i.

$$w_i \rightarrow GloVe(w_i) \rightarrow \tilde{w_i} \quad (1)$$

4.2 Emoji Encoder

We first extract the emojis from sentences and pass them through certain filters so as to remove redundant characters and retain only the emojis. This is achieved with the help of a python library called Enchant. The embeddings for the filtered list of emojis are retrieved using emoji2vec. These embeddings are then

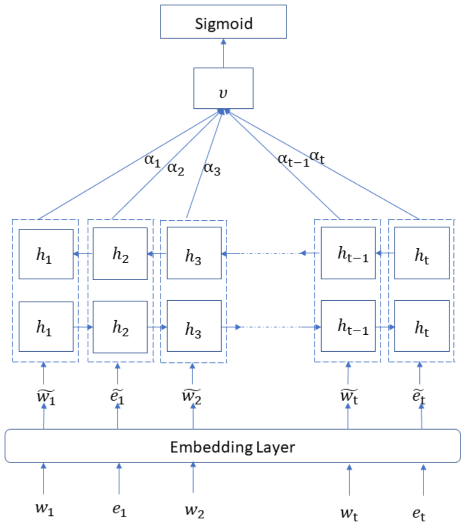

Fig. 5. The proposed framework ESD for sarcasm detection takes a list of words $[w_1, w_2 ... w_t]$ and emojis $[e_1, e_2, ..e_t]$ as input and converts them into word $[\tilde{w_1}, \tilde{w_2}, ...\tilde{w_t}]$ and emoji $[\tilde{e_1}, \tilde{e_2}, ...\tilde{e_t}]$ embeddings. $[h_1, h_2, ...h_t]$ denotes the list of concatenated vectors which are passed through the bi-directional GRU. The attention weights $[\alpha_1, \alpha_2, ...\alpha_t]$ are then multiplied and summed with the vector representations to give the context vector, υ. This vector υ is finally passed to the sigmoid function for classification.

stored in the embedding matrix W along with the word embeddings. The emoji embeddings play a major role in determining sarcasm. These embeddings help us in understanding complex emotions which cannot be derived from words alone. Given a list of sentence vectors S_i, for every emoji e_i, there exists an emoji embedding $\tilde{e_i}$:

$$e_i \rightarrow emoji2vec(e_i) \rightarrow \tilde{e_i} \tag{2}$$

4.3 Sarcasm Detection

Once the embedding matrix W is complete, we pass these embeddings as an input to a bi-directional GRU. The embedding $\tilde{s_i}$ could either be a word embedding $\tilde{w_i}$ or an emoji embedding $\tilde{e_i}$. We utilize the bi-directional GRU to obtain a forward hidden state $\overrightarrow{h_{it}}$ and a backward hidden state $\overleftarrow{h_{it}}$. The concatenated vector $h_i = [\overrightarrow{h_{it}}, \overleftarrow{h_{it}}]$ represents the information of the whole sentence centered around w_i .

$$\overrightarrow{h_i} = \overrightarrow{GRU}(\tilde{s_i}), i \in [1, T] \tag{3}$$

$$\overleftarrow{h_i} = \overleftarrow{GRU}(\tilde{s_i}), i \in [T, 1] \tag{4}$$

Our model uses an embedding layer of 300 dimensions for mapping both words and emojis to vector space. We use the bidirectional GRU with dropout, which takes the vector representation of each word and emoji in the dataset as an input. Finally, an attention layer takes the previous layers as input and weighs

Table 1. The statistics of datasets

Datasets	Twitter	Facebook
No. of Sarcasm	$6,592$	$2,668$
No. of Non-sarcasm	$10,267$	$2,803$

each word according to its importance or relevance in the text. The Activation function, Sigmoid and Adam optimizer are used to build the model.

$$u_i = tanh\,(Wh_i + b_w) \tag{5}$$

$$\alpha_i = \frac{\exp(u_i^T u_w)}{(\sum_i \exp(u_i^T u_w)} \tag{6}$$

$$\upsilon = \sum_i \alpha_i h_i \tag{7}$$

Here, u_i is the score obtained by applying a hyperbolic tangent function over the product of h_i, the vector representation of the word, and the weight matrix W, and a bias b_w is added to the product. This score is generated for each state in the bi-directional GRU. υ is the final representation vector of the text.

υ is also referred to as the context vector. υ is the weighted summation of the product of attention weight α_i and word representation h_i. These context vectors are computed for every word and are given as an input to the final sigmoid layer for classification.

5 Experiments

In this section, we present the experiments to evaluate the effectiveness of the proposed ESD framework. Specifically, we aim to answer the following evaluation questions:

- Is ESD able to improve the sarcasm detection performance by modeling text and emoji information simultaneously?
- How effective are the text and emoji features, respectively, in improving the sarcasm detection performance of ESD?

5.1 Datasets

Data was collected from Twitter for a 5-month period and from Facebook for two years from 2015 to 2017 using web scraping in Python [3]. The sarcastic pages such as ‘sarcasmLOL’, ‘sarcasmBro’ from Facebook and tweets with hashtags, ‘sarcasm’, ‘sarcastic’ were taken from Twitter. The data was scraped, preprocessed and the data containing only text plus emoji were extracted. The data preprocessing involved removal of hyperlinks, special characters, hashtags, retweets, etc. The statistics of the datasets are given in Table 1.

[3] https://github.com/jsubram/Sarcasm-Detection-Using-Emoji

5.2 Comparison Of Sarcasm Detection Methods

The representative state-of-the-art sarcasm detection methods that are compared with ESD, are listed as follows:

- FSNN [3]: FSNN stands for Fracking Sarcasm using a Neural Networks, which uses a Convolutional Neural Network (CNN) followed by an LSTM and a Deep Neural Network (DNN) to detect sarcasm in a sentence.
- CASCADE [6]: CASCADE stands for Contextual Sarcasm Detection in Online Discussion Forums. CASCADE uses CNNs to capture the user's personality features to boost the performance of classification.
- RCCSD [4]: RCCSD stands for The Role of Conversation Context for Sarcasm Detection, which uses conditional LSTM networks with sentence-level attention on conversational context and response.

Table 2. Best performance comparison for Sarcasm detection

Datasets	Metric	FSNN	CASCADE	RCCSD	ESD
Twitter	Accuracy	0.891	0.753	0.763	0.991
	Precision	0.910	0.798	0.768	0.998
	Recall	0.904	0.802	0.791	0.976
	F1	0.899	0.867	0.820	0.987
Facebook	Accuracy	0.878	0.745	0.768	0.971
	Precision	0.901	0.771	0.733	0.975
	Recall	0.889	0.789	0.745	0.979
	F1	0.893	0.842	0.772	0.969

5.3 Performance Comparison

We compare ESD with state-of-the-art sarcasm detection methods. The metrics for evaluation are Precision, Recall, F1 and Accuracy. The dataset is split into training, testing, and validation in the ratio 6:2:2. All models are trained for 50 epochs with early stopping and their results are shown in Table 2.

- In general, ESD outperforms other baselines. We use word embeddings, to learn the representations of each word, and emoji embeddings to learn complex sentiments in the sentence that are not easily learned by word embeddings alone. We also add an attention layer to focus on the part of the sentence which has sarcasm, thereby enhancing our model's performance.
- FSNN has the highest Accuracy, Precision, Recall and F1 score amongst all the three baseline models. This is due to the fact that the model architecture of FSNN is much deeper and more complex than the other two models. The good results show that the depth of the neural network helps in the better learning of word representations
- CASCADE performs slightly better than RCCS because it utilizes CNN to capture complex stylometric and personality features of the user. The results demonstrate the importance of extracting features from sentences in detecting sarcasm.

– RCCSD uses an Attention-based LSTM to model both context and response. The key feature of RCCSD is its attention layer. The attention is used to identify the sarcastic part in the response. This indicates that having an attention layer on top of an LSTM can help our model to focus on the part of the sentence which contains sarcasm.

Table 3. The results of average F1 scores

Datasets	Classification Algorithm	Text	Emoji	Text+Emoji
Twitter	SVM	0.7674	0.8288	0.8465
	Decision Tree Classifier	0.7430	0.8463	0.8803
	Random Forest	0.7773	0.8522	0.8729
	Adaboost Classifier	0.7690	0.8466	0.8964
	Gradient Boosting Classifier	0.7821	0.8549	0.8876
	K Neighbors Classifier	0.7699	0.7376	0.8934
	Stochastic Gradient Descent	0.7804	0.8635	0.8810
	Bayesian Classifier	0.7557	0.8454	0.8943
	ExtraTreesClassifier	0.7755	0.8518	0.8936
Facebook	SVM	0.6496	0.9283	0.9302
	Decision Tree Classifier	0.6460	0.9523	0.9432
	Random Forest	0.6809	0.9596	0.9615
	Adaboost Classifier	0.7023	0.9578	0.9578
	Gradient Boosting Classifier	0.7223	0.9597	0.9670
	K Neighbors Classifier	0.5429	0.8093	0.8751
	Stochastic Gradient Descent	0.6361	0.9561	0.9506
	Bayesian Classifier	0.6497	0.9189	0.9287
	ExtraTreesClassifier	0.6865	0.9523	0.9597

5.4 Assessing Text and Emoji Components

The Text and Emoji Components are obtained as embeddings using the Word2Vec and Emoji2Vec methods. We test our baseline features on 9 different widely used machine learning algorithms such as Support Vector Machines(SVM), Decision Tree Classifier, etc. From Table 3, we observe the following:

– Only word embeddings: When we train our model with only word embeddings, we observe that the model struggles to learn sarcastic features in the data since it is difficult to infer sarcasm using only words.
– Only emoji embeddings: When we train our model with only emoji embeddings, we observe that the model performs better than it performed with word embeddings. This is because emojis are able to convey complex emotions that are essential to detect sarcasm.
– Both word and emoji embeddings concatenated: The word embeddings and emoji embeddings are concatenated horizontally and are given as input to the model. We observe that the model is able to perform considerably better than it did with only word and only emoji embeddings because the model is able to relate complex emotions with the contextual meaning. This enables the model to detect sarcasm more accurately.

6 Conclusion

Emojis provide a new dimension to social media communication. We study the role of emojis for sarcasm detection on social media. We propose a new deep learning model by introducing an attention layer which helps to model the text and emojis simultaneously for sarcasm detection. The empirical results on real-world datasets demonstrate the effectiveness of the proposed framework.

Acknowledgements

This material is based upon work supported by, or in part by, the ONR grant N00014-17-1-2605 and N000141812108.

References

1. Barbieri, F., Kruszewski, G., Ronzano, F., Saggion, H.: How cosmopolitan are emojis?: Exploring emojis usage and meaning over different languages with distributional semantics. In: Proceedings of the 2016 ACM on Multimedia Conference. pp. 531–535. ACM (2016)
2. Eisner, B., Rocktäschel, T., Augenstein, I., Bošnjak, M., Riedel, S.: emoji2vec: Learning emoji representations from their description. arXiv preprint arXiv:1609.08359 (2016)
3. Ghosh, A., Veale, T.: Fracking sarcasm using neural network. In: Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis. pp. 161–169 (2016)
4. Ghosh, D., Fabbri, A.R., Muresan, S.: The role of conversation context for sarcasm detection in online interactions. arXiv preprint arXiv:1707.06226 (2017)
5. Hallsmar, F., Palm, J.: Multi-class Sentiment Classification on Twitter using an Emoji Training Heuristic. Master's thesis, KTH Royal Institute of Technology School of Computer Science and Communication (05 2016)
6. Hazarika, D., Poria, S., Gorantla, S., Cambria, E., Zimmermann, R., Mihalcea, R.: Cascade: Contextual sarcasm detection in online discussion forums. arXiv preprint arXiv:1805.06413 (2018)
7. Hu, X., Tang, J., Gao, H., Liu, H.: Unsupervised sentiment analysis with emotional signals. In: Proceedings of the 22nd international conference on World Wide Web. pp. 607–618. ACM (2013)
8. Joshi, A., Bhattacharyya, P., Carman, M.J.: Automatic sarcasm detection: A survey. ACM Computing Surveys (CSUR) 50(5), 73 (2017)
9. Kelly, R., Watts, L.: Characterising the inventive appropriation of emoji as relationally meaningful in mediated close personal relationships (2015)
10. Lee, H., Kwak, N.: The affect effect of political satire: Sarcastic humor, negative emotions, and political participation. Mass Communication and Society 17(3), 307–328 (2014)
11. Morstatter, F., Shu, K., Wang, S., Liu, H.: Cross-platform emoji interpretation: analysis, a solution, and applications. arXiv preprint arXiv:1709.04969 (2017)
12. Novak, P.K., Smailović, J., Sluban, B., Mozetič, I.: Sentiment of emojis. PloS one 10(12), e0144296 (2015)
13. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: EMNLP. vol. 14, pp. 1532–43 (2014)
14. Rajadesingan, A., Zafarani, R., Liu, H.: Sarcasm detection on twitter: A behavioral modeling approach. In: Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. pp. 97–106. ACM (2015)