Predicting Topic Change and Emoji Usage from Twitter Data

Michał Bień, Kevin Guyard, Yiming Li

December 2020

Abstract

The research on human social interaction on twitter has become very active in the recent years, especially in the area of computational linguistics. Most research works so far focused on the tweets' content, sentiment and the associated topic. However, an important part of this messaging schema had remained mostly unnoticed: the emojis. The extensive use of these Unicode characters only recently gained the attention of researchers, leading to the first publications on the association of the message topic to certain emoji. In our work, we study, whether emojis play an important role in topic change in turn-taking twitter conversations. We perform the emoji classification task to show, how the use of certain emojis is connected to the written message context. We create a new twitter conversation dataset with a subset annotated for topic change.Finally, we test supervised and unsupervised machine learning methods for topic change detection to evaluate, how important are the emoji for triggering a topic change.

1 Introduction

The use of emojis has been prevalent in social media for the past decades. In 2015 the Oxford Dictionaries chose, for the first time, an emoji as the word of the year. In fact, numerous studies have examined the emoji interpretation, cluster analysis and emotion detection across multiple disciplines such as natural language processing (NLP), marketing and behavioral science[18, 1, 3, 7, 4]. Emojis as non-verbal cues, serve to emphasize or de-emphasise the interpretation of the speech act and build connections in conversations[6].

Research on analyzing the meaning of emojis and relation of emojis with their textual contents has been sparse. Emojis seem to be associated with mitigated/aggravated emotions, sarcasm markers and intention elucidation[21, 19, 5]. Although emojis are Unicode characters, misunderstandings might still occur due to variations in emoji interpretations and multiple chatting platform designs[20]. It is a pressing need for us to investigate contextual information related to emojis. Fortunately, the machine learning models provide a uniquely suited tool for studying such a topic in a systematic manner. More precisely, our study examines the performance of emoji classification method using long short-term memory (LSTM)[8] recurrent neural network (RNN) architecture, on the Twitter textual dataset.

Emojis are surrogates for nonverbal behaviors in online written communication that allow to manage turn-taking behaviors in conversations[5, 10]. There is a dearth of research work addressed to the question whether emojis facilitate filling the communication gap and clarifying message intentions to begin topic transitions in online written communication. Meanwhile, finding a method to more appropriately detect topic transition in online written conversations remains an unsolved challenge. Given the possibility of biased outcomes from manual annotation in topic change, our study aims at using unsupervised and supervised machine learning methods to detect, how topic changes happen in turn-taking conversations.

2 Twitter Data

2.1 Data acquisition

To acquire the necessary data to train the machine learning models, we conducted a two-month tweet data collection via Twitter API and *Tweepy*[17] package for Python. By using web-scraping techniques we were able to obtain tweets with emojis and process them into a conversational dataset. The data collection was slow, due to Twitter's measures to limit automated access to tweets and the scarcity in conversation dataset with emojis from online database.

Our first dataset was composed of 1 million tweets with emojis. The key purpose of this dataset was to examine whether there is a non-negligible relation between the emoji used, and the context of the tweet. We were aiming to see what the nature of this relation is. The second dataset contained 12,000 conversation samples which consisted of 3 following replies, one causal for another. To generate this dataset, we constructed a reverse hierarchical relation through retrieving features namedin_reply_to_status_id_str, id_str and in_reply_to_screen_name with Twitter API. In the first procedure, we applied the same method as in first dataset to get root node tweet full_text and their corresponding in_reply_to_status_id_str and in_reply_to_screen_name features from json files. After that, we searched for last ten tweets on user_timeline by querying in_reply_to_screen_name from root node and kept subsequent tweets when in_reply_to_status_id_str(first procedure) and id_str(second procedure) matched in the dataset.

As Twitter API imposes restrictions on the scope of tweet reply search, this limited our research work to consider three-turn conversation segment from the full conversation cascade. After the dataset was ready, we filtered out conversations without emojis in order to investigate the topic change detection for tweets accompanied with emojis.

2.2 Data preprocessing

We followed two approaches to preprocess our datasets, given that we had two different tasks with respect to both emoji prediction and topic change detection.

In the first dataset, we separated the text from corresponding emojis as a necessary step for the further prediction task. Since irrelevant contents might have affected the final accuracy rate, we removed all name mentions, hyperlinks and empty texts. As some tweets contained more than one emoji, we proceeded to get one dominant emoji in each tweet by filtering out less frequent emojis in raw text to reduce noise in the labels. Next, we performed the filtering step to keep tweets with top 20 popular emojis to enhance predictive performance of our model. Our preprocessing procedure for the first dataset did not remove stop words (like to, of, the, etc.) because it was assumed that tweets without stop words might affect textual interpretations and tweets are informal texts from social media. Inspired by Felbo et al.'s [7] research work, we expected that tweets with emojis can be used as training set for models, as emoji itself can be regarded as annotated label for tweets. Then, we have used a text to sequence method to transform the text of tweets to a useful matrix to ML process. Thus, 80% dataset were chosen for training and 20% for evaluating the model performance.

In the second dataset, the additional step of stopwords removal and text lemmatization was performed. It was assumed, that the topic of the conversation should be highly associated with the specific keywords and therefore it was useful to both remove the "useless" words and put emphasis on the "useful" keywords by lemmatizing them.

3 Emoji Prediction

As Luda Zhao and Connie Zeng suggested in their research work[22], a recurrent neural network with LSTM layers can be used to process emojis classification task with good accuracy. This neural network is fed by an embedding vector of size x (see Table 1 for the design of the embedding size)

In order to design an efficient neural network, we performed a grid search for identifying the optimal hyperparameters. To determine the number of layers, we tried with a single, two, and three stacked LSTM layers. We found out that the best results were given by single-layer LSTM. Then we studied several activation functions for the network and we retained the 4 following ones, since their result were similar: softplus, softmax, sigmoid and exponential.

For these four types of activation function, we tuned the hidden sizes of the LSTM layer in the set $\{50, 100, 200\}$ and the embedding sizes of the input vector in the set $\{20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 250, 300, 350\}$. After this step, we iterated over the dropout probability into the range 0% to 90% by step increment of 10%.

The last step was to explore the penalization of the edges weights with a L2-regularization. We iterated over the trade-off parameter into the range 1e-6 to 1 by log step of 10. However, none of the value of the trade-off parameter for the regularization significantly increased the accuracy of the model. The different hyperparameters found are shown in Table 1. We note, however, that hyperparameters differ from an activation function to another.

	Hidden	Embed.	Dropout	Acc.
Softplus	100	80	0.2	13.7
Softmax	100	180	0.5	13.8
Sigmoid	100	160	0.3	13.4
Exponential	100	60	0.2	13.6

Table 1: Activation Function

Finally, the best recurrent neural network for this task is a one LSTM layer of hidden size 100 with an embedding size of 180, a dropout probability of 50% and a sigmoid function as activation function in the last layer.

In order to increase the accuracy, we collected more tweets before the final testing phase. Thereby, we passed of a 400K tweets design dataset to a $1.5 \cdot 10^6$ tweets final train/test dataset (These indicated number are before cleaning). The finals results are given in Table 2

# Emojis	5	10	15	20	25
Accuracy	0.381	0.258	0.202	0.178	0.167

Table 2: Final result for RNN

4 Topic change

The second part of our research was aimed at answering the question: **Can emojis included in a tweet influence the topic change?**. First, we generated ground truth data using manual data subset annotation. We then tested two unsupervised topic clustering techniques: Latent Dirichlet Allocation(LDA) [2] and K-Means. The topic changes were deduced from the difference between the clustered topics inside the same conversation. We also used a manually labelled training set for training a supervised solution: a LSTM model which served as the topic change detector. Both unsupervised and supervised topic detection algorithms were then evaluated against human-generated ground truth labels.

4.1 Ground truth data

To evaluate the methods properly, we first generated the ground truth labels for the conversation data which was previously gathered from twitter. Each conversation was annotated with two labels: as there were three messages in each conversation, each label corresponded to a transition between the adjacent messages. If the two messages had a different main topic, they were assigned "change" label. Other possible labels were: "no change" and "unsure".

To generate the data, a custom text annotation software was created. The task was performed by 3 annotators, who labeled 300 samples of discussion (600 transitions) in total. In addition to labelling full-tweet topic change, the annotators were also asked to guess the topic transition based only on the emojis included in the tweets. This allowed us to assess, how hard this task is expected to be for the supervised model. Out of the collected data, 84.5% was conclusive (evaluator didn't specify "unsure" as a transition label).

The emoji-only topic change was evaluated against the conclusive samples from the full-tweet topic change data (referred as ground truth from now on). The 81% of the responses were conclusive. The accuracy of detection the change was 56.1%, and the F1 score 0.33. Given the binary classification task, the result shows that the difficulty of emoji-only topic change assessment is an extremely hard task for the human evaluator. Therefore, we decided to refrain from testing the emoji-only approach using the machine learning methods.

4.2 Unsupervised topic learning

Topic detection on the textual data has been an area of intensive research in the recent years. Multiple clustering methods use the intrinsic differences between the distinct text samples to segregate them into distinct topics in the unsupervised manner. We used and evaluated two different approaches to annotate topic change using clustering methods. We assumed that, if the clustering algorithm indicates two topics of the adjacent messages as different, these two messages' contents should be different enough to indicate topic change.

We used LDA method to cluster the topics on top of vectorized token-count representation of the complete tweet texts. The implementations of *LatentDirichletAllocation* and *CountVectorizer* provided by SciKit-Learn[15] were used for this purpose. Using the algorithm, each message in the three-turn conversation was assigned a specific topic. Next, those topics were compared, and each difference between the two adjacent messages was labelled as topic change.

This approach was evaluated against the manually labeled data, after performing a hyperparameter search - LDA topic and epoch numbers. Out of the test set, half was used for hyperparameter tuning and the remaining part - for the final evaluation. The obtained results are reported in Table 3. For each number of topics, only the best performing number of epochs is presented.

Topics	Iterations	F1
30	5	0.503
40	5	0.524
50	5	0.523
60	20	**0.528**
70	5	0.520

Table 3: Results of LDA method for unsupervised topic change

As the score of naive classifier on our test set is 0.51, we can consider LDA method as yielding no significant advantage for our case. It's possible that the amount of data available in each message was not enough for the LDA, which usually operates on the whole documents, to make a reasonable document statistics.

The second approach leveraged an idea of topic clustering based on document vectors [13]. We extracted word vectors [12] from pretrained SpaCy [9] model and calculated the mean document vector for each tweet. These 300-features vectors were then fed into *KMeans* unsupervised learning algorithm using the implementation provided by SciKit-learn. The results of using this method are shown on Table 4.

Topics	Max Iter.	F1
2	100	0.259
10	100	0.415
50	**100**	**0.464**
100	300	0.169

Table 4: Results of wordvec clustering method for unsupervised topic change

The main difference between this approach and LDA is the algorithm's lexical base. Specifically, LDA uses only the lexical resources found in the dataset to propose the clustered topics. On the other hand, the word vectors provided by SpaCy provide much larger context - the vectors are trained on the whole Common Crawl dataset using GLoVE [16] technique.

Apparently, the method is performing worse than the LDA. In fact, its predictive performance is even worse than a naive classifier, which may lead to the conclusion, that the GloVE vectors are not useful in this context, or should be used in different manner.

4.3 Modeling the change

In this section, we present our approach to use supervised learning in form of LSTM Networks[8] to perform topic change detection from input data.

This approach provided the model with all the content of the three following messages of the twitter conversation, and expects it to predict the topic change. The purpose is to investigate, how well this training approach can model a topic change. The LSTM was trained using pyTorch [14] machine learning library supported by Adam optimizer [11]. Due to the modest amount of labelled data, our data was split as follows: 200 samples of training set, 50 samples of dev set and 50 samples of test set. The results of training on full data are displayed on Table 5.

The better score of LSTM supervised method than the unsupervised methods may lead to a conlusion, that human understands a topic change in a different manner than machine do. More precisely, the human can perceive the topic change where the topic is the same from the clustering perspective, and vice versa.

Hidden	Epochs	Embedding	LR	F1
32	60	300	0.01	0.558
64	60	300	0.01	0.586
128	**30**	**300**	**0.01**	**0.598**
192	30	300	0.01	0.593

Table 5: Result of LSTM topic detection on full data

5 Conclusions

While our novel approach to detecting the topic change seem promising, we still need to further optimize our algorithms to achieve better model emojis prediction and topic change. The evaluation results did not satisfy us, and we observe multiple reasons for this.

Regarding the emoji classification task, the human language itself has multiple metaphors and implicit information, which create barriers when we try to model textual information with emoji usage. Future studies could include attention model to predict emoji usage from specific aspects in the textual information (such as sarcasm, sentiment and semantic meaning). Moreover, the emojis use differs from a person to another. Because the model is not the same for everyone, another features need to be took into consideration like age, culture and also friend since human groups influence each person of these groups. In addition, we need better well-tuned and fine-grained algorithms for emoji prediction tasks. Particularly, our task only focuses on single label for emoji use for each tweet. Question remain unanswered if we consider multi-label for such emoji usage task.

To improve the results of the topic change task, above all, the number of our samples, both labeled and unlabeled, have to be increase. Also, we need standardized criteria for topic change annotation to minimize the chances of ground truth data being ambiguous and biased. Further research could employ more native speakers from Amazon Mturk to reduce biased annotation results. Moreover, we only selected segment of conversations (three-turn) from full conversations on Twitter. It would be advisable to acquire the complete conversation string. Without having the full conversation data, topics might be incoherent, and turn adjacency can also be disrupted in multi-party turn-taking conversations.

While our work shows the potential of using both supervised and unsupervised method to approach the problem of topic change detection, further research work is required better to model the difference between topic changes for two-party and multi-party turn-taking conversations.

References

[1] Francesco Barbieri, Francesco Ronzano, and Horacio Saggion. “What does this emoji mean? a vector space skip-gram model for twitter emojis”. In: *Calzolari N, Choukri K, Declerck T, et al, editors. Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016); 2016 May 23-28; Portorož, Slovenia. Paris: European Language Resources Association (ELRA); 2016. p. 3967-72.* ELRA (European Language Resources Association). 2016.

[2] D. Blei, A. Ng, and Michael I. Jordan. “Latent Dirichlet Allocation”. In: *J. Mach. Learn. Res.* 2003.

[3] Aditi Chaudhary et al. “What A Sunny Day: Toward Emoji-Sensitive Irony Detection”. In: *W-NUT 2019* (2019), p. 212.

[4] Maureen A Coyle and Cheryl L Carmichael. “Perceived responsiveness in text messaging: The role of emoji use”. In: *Computers in Human Behavior* 99 (2019), pp. 181–189.

[5] Henriette Cramer, Paloma de Juan, and Joel Tetreault. “Sender-intended functions of emojis in US messaging”. In: *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services.* 2016, pp. 504–509.

[6] Eli Dresner and Susan C Herring. “Functions of the nonverbal in CMC: Emoticons and illocutionary force”. In: *Communication theory* 20.3 (2010), pp. 249–268.

[7] Bjarke Felbo et al. “Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm”. In: *arXiv preprint arXiv:1708.00524* (2017).

[8] S. Hochreiter and J. Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9 (1997), pp. 1735–1780.

[9] Matthew Honnibal and Ines Montani. “spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing”. To appear. 2017.

[10] Linda K Kaye, Helen J Wall, and Stephanie A Malone. ““Turn that frown upside-down”: A contextual account of emoticon usage on different virtual platforms”. In: *Computers in Human Behavior* 60 (2016), pp. 463–467.

[11] Diederik P. Kingma and Jimmy Ba. “Adam: A Method for Stochastic Optimization”. In: *CoRR* abs/1412.6980 (2015).

[12] Tomas Mikolov et al. “Efficient Estimation of Word Representations in Vector Space”. In: *CoRR* abs/1301.3781 (2013).

[13] Guilherme Raiol de Miranda, Rodrigo Pasti, and Leandro Nunes de Castro. “Detecting Topics in Documents by Clustering Word Vectors”. In: *Distributed Computing and Artificial Intelligence, 16th International Conference.* Cham: Springer International Publishing, 2020, pp. 235–243. ISBN: 978-3-030-23887-2.

[14] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32.* Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 8024–8035.

[15] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[16] Jeffrey Pennington, R. Socher, and Christopher D. Manning. “Glove: Global Vectors for Word Representation”. In: *EMNLP.* 2014.

[17] Joshua Roesslein. *Tweepy.* 2020. URL: `https://www.tweepy.org/` (visited on 12/13/2020).

[18] Leah Warfield Smith and Randall L Rose. “Service with a smiley face: Emojional contagion in digitally mediated relationships”. In: *International Journal of Research in Marketing* 37.2 (2020), pp. 301–319.

[19] Dominic Thompson and Ruth Filik. “Sarcasm in written communication: Emoticons are efficient markers of intention”. In: *Journal of Computer-Mediated Communication* 21.2 (2016), pp. 105–120.

[20] Garreth W Tigwell and David R Flatla. “Oh that’s what you meant! Reducing emoji misunderstanding”. In: *Proceedings of the 18th international conference on human-computer interaction with mobile devices and services adjunct.* 2016, pp. 859–866.

[21] Ilona Vandergriff. “Emotive communication online: A contextual analysis of computer-mediated communication (CMC) cues”. In: *Journal of Pragmatics* 51 (2013), pp. 1–12.

[22] Luda Zhao and Connie Zeng. “Using neural networks to predict emoji usage from twitter data”. In: *Sementic Scholar* (2017), pp. 1–6.