Emojional: Emoji Embeddings

Elena Barry, Shoaib Jameel, and Haider Raza

School of Computer Science and Electronic Engineering, University of Essex, UK,
{eb20068, shoaib.jameel, h.raza}@essex.ac.uk

Abstract. Emojis are intended to illustrate or replace words in natural language. Although they are understandable across linguistic barriers, their implications are ever-evolving, and change depending on who we are talking to, and when. The underlying emotion is key. This is evident in language online, highlighted recently by the rise of cancel culture via online shaming. For example, the use of the clown emoji 🤡 to signify someone is making a fool of themself, or the collective spamming of the snake emoji 🐍 to "cancel" someone, both show seemingly innocent emojis being used as clear forms of aggression online. To capture these nuances, we have created novel emoji embeddings trained on their emotional content. The subsequent emoji embeddings are generally more accurate than the state-of-the-art embeddings on the task of sentiment analysis. These embeddings can be found in our GitHub repository: https://github.com/elenabarry/emojional.

Keywords: emoji, emoji embeddings, emotion, machine learning

1 Introduction

An effective language model should possess the ability to not only model text, but also a range of other instance types, such as multi-media content and others which are popularly used by social media users. One of the commonly occurring instances are emojis which are often ignored when processing data emanating from social media. Emojis play a central role in the effective modelling of users on social media, for instance, in [17], the authors used emojis as one of the features to model depressed users online where emojis were found to be one of the most important modelling features. Therefore, modelling emojis effectively is an important problem to address.

Computers do not naturally understand words or emojis, which play a central role in user interactions in online social networks. One of the popular ways to encode this data on the web is by representing it numerically in low-dimensional vector space. By encoding the data in a numeric form, mathematical rules can be applied to them as shown using conceptual spaces [5]. These encodings are called embeddings and can be used as inputs to a downstream model in several different applications.

A word or emoji embedding is a vector representation of that word or emoji. According to the distributional hypothesis, [9], words and emojis with similar

meanings will have vector representations that are close together in the embedding space. The goal of the embedding space is to learn a spatial relationship between words and emojis which will correspond to their meaning or context.

By pre-training word and emoji embeddings, they can then be applied to different natural language tasks such as sentiment classification and depression detection [17]. This simplifies machine learning as words and emojis do not need to be trained for each task; these general representations can be used across tasks. The embeddings can be constructed in different ways to produce different spatial relationships, for example, by using different datasets, or by using dictionary definitions to map the distributional relationship.

While static embeddings have been dominant in the past, in recent times, contextual language models such as Bidirectional Encoder Representations from Transformers (BERT) [6] have shown state-of-the-art results in various natural language processing (NLP) and other domains such as computer vision. These models are usually trained in the domain-independent environment, for instance, BERT has been trained on Wikipedia and BookCorpus which enables them to capture various intricate properties in the text which includes faithfully modelling the syntactic and semantic nature of the natural language text. These models are then fine-tuned in their domain-specific tasks under a transfer learning setting which makes them reliable to model domain-specific problems. However, these language models do not naturally encode emoji embeddings.

Currently, to include emoji embeddings in these models, we either need to fine-tune a contextual model with emojis which usually requires a labelled emoji dataset, or we need to use separate pre-trained static embeddings, for example, "emoji2vec" [7], which were learned from their "Unicode" description and keywords. Their method mapped emojis into the same space as the 300-dimensional Google News word2vec embeddings [12], allowing them to be used readily alongside word embeddings.

Emojis are a major component of text online, and yet, they are often omitted in NLP. This lack of emoji support could affect performance on downstream tasks. The younger demographic tends to use more emojis and less text and so there is an increasing importance in having effective emoji embeddings, however there are various challenges when modelling emojis. For instance, new emojis are regularly introduced and they are not natural language text. This changing landscape of the emoji makes it time-consuming to create new emoji datasets which they are often sparsely distributed across. In contrast to other research, we have developed new emoji embeddings or emoji vector representations that uses a more fluid approach to allow for a wide range of emotional content for the ever-evolving emoji.

2   Literature Review

Emoji embeddings have been either created by embedding the emojis directly, for example, Barbieri et al., [3], trained emoji embeddings from a large Twitter dataset of over 100 million English tweets using the skip-gram method [12], or

indirectly, considering emojis as words upon which the embedding will be based. For example, the most prominent emoji embeddings, emoji2vec [7] are trained on emoji phrases and keywords learnt from their Unicode description, similar to Hill et al. [10], who developed word representations for words based on their dictionary meaning. Emoji2vec obtained more accurate results than its large dataset counter parts at the time, such as Barbieri et al. [3]. This may be due to the corresponding text in the dataset skewing the results, as even when emojis are embedded directly, they are embedded as words in a sentence rather than as separate emotional entities. Ai et al. [2] like emoji2vec created sense_des emoji embeddings, using possible meanings and senses associated with their emoji description. However, the emoji2vec embeddings do not capture the full emotional characteristics of an emoji as its associated keywords do not always include emotive words, and since 2016, there have been many more emojis added. There are currently 1816 emojis in v13.1 of Unicode[1]. This illustrates the fast development and evolving language of the emoji, which needs to be trained quickly for the models to remain relevant.

Our results differ from emoji2vec [7] because the information used to train each emoji is different. We view emojis as a tonal device full of emotional information. For example, the "clown face" emoji 🤡, which is usually used to convey foolishness, selfishness, idiocy or that something is creepy or scary is trained using these key emotional words. The same emoji is trained with the phrases "scary clown" and "clown face" in emoji2vec [7]. Although emoji2vec captures some of the emoji's emotional content from the word "scary" in the phrase "scary clown", it does not manage to capture it all. In natural language tasks, such as sentiment analysis and emotion prediction, the key information needed from the emoji is its emotional content.

Chen et al. [4] proposed emoji embeddings that can handle situations when the semantics, or the sentiment of the emojis contradicts the information conveyed in the text. To do this, the authors created a bi-sense embedding where each emoji is embedded into two distinct vectors, a positive sense and a negative sense. This allows for learning different representations of semantic and sentiment information. Although this method is effective for sentiment analysis, grouping emojis as either negative or positive is very limiting for other natural language tasks, as it loses emotional values, which could be helpful in natural language tasks, such as sarcasm detection.

Guibon et al. [8] created emoji embeddings directly from real usage. The authors then used these embeddings to create emoji clustering to automatically identify groups of emojis. Their method used the unsupervised learning technique, a continuous bag of words (CBOW) [12], to extract patterns and knowledge from real emoji usage. This consisted of analysing the similarity of context usage between emojis. First, word embeddings were applied on tweets containing emojis, then these embeddings were applied to a spectral clustering algorithm. This clustering allows better emoji prediction as it enables the capturing of a sense or a feeling, such as "happy". Their embeddings are called Face Emojis.

---

[1] https://unicode.org/emoji/charts/full-emoji-list.html

This work produced state-of-the-art results, but the method assumes that emojis are part of written natural language and can be considered as words.

Ahanin and Ismail [1] take a different approach and created the embeddings called FuzzyMoji2Vec. Unlike the existing emoji embedding approaches that were constructed based on the emoji description, emoji, and text correlation, or spectral clustering, they used the correlation of emoji and emotion labels to classify emojis into one or more emotion classes. Then they created emoji embeddings based on their assigned classes. A smaller quantity of data was required to train the emojis, and the labelling is automatic. To cluster the emojis into emotion classes, they used fuzzy clustering [16], a machine learning algorithm in which each data point can belong to more than one cluster. To perform this, they used a dataset that included tweets that were labelled into the 11 classes of Plutchiks theory of emotion [14] and collected new data from Twitter which they labelled automatically using the already classified dataset.

This clustering technique shows that emojis have the potential of multiple emotional layers, allowing emojis to represent nuanced sentiment. In the last step of their processing, the authors used Plutchik's notion of emotional opposites. For example, in Plutchik's theory, joy is the opposite of sadness, and therefore in their approach, an emoji can be either in a joy or sadness class but not both. This does not accurately embed emojis such as the popular "loudly crying face emoji"[2] which according to the "Emojipedia"[2] description is used to "...convey inconsolable grief but also other intense feelings, such as uncontrollable laughter, pride or overwhelming joy". Therefore, their model cannot always capture the full range of emoji emotion. These constraints illustrate that Plutchik's notion of emotion is too narrow to map emojis.

Our work is substantially different, for instance, none of the previously mentioned research has modelled a wide emotional range in emoji embeddings. We have created novel emoji embeddings that have multifaceted emotions, feelings and sensations that are not constrained to categories. The model is not dependent on large scale datasets, therefore, infrequent emojis are modelled as effectively as the most frequently occurring emojis. This also means that the model can be adapted as new emojis develop and current emojis evolve. This makes our model effective and durable in a range of different natural language processing tasks.

3   Our New Emoji Embeddings or Representations

As highlighted in Section 2, there are only a few recent approaches that have modelled emoji embeddings. The fundamental problem is that shortly after the emojis are created, their embeddings go out of date as new emojis are added every year, for instance, 217 emojis were added in 2021 according to Emojipedia[3], and over time emoji meanings evolve, for example, the raised fist emoji ✊ gained political nuance in 2021 in contrast to its perception in previous years.

---

[2] https://emojipedia.org/loudly-crying-face/

[3] https://blog.emojipedia.org/217-new-emojis-in-final-list-for-2021/

This versatile nature of the emoji means that even when large amounts of emoji rich data are present, it is still difficult to pinpoint their underlying meaning. Firstly, this could be due to the higher probability that an emoji can appear anywhere within a sentence, as it is not constrained to the same grammar rules as words, therefore, we would need a much larger dataset to embed emojis to capture all their semantic meanings. Secondly, emojis are not used liked words in sentences, they are mainly used to demonstrate the tone of the text and are full of emotional information, and therefore, can be successfully embedded as emotional identities.

With the current rate of the development of new emojis and their changing usage, an adaptive model is essential. Until emoji meanings and representations settle down and are not so volatile to new interpretations, a dictionary-based model is cheaper to maintain and update as emoji meanings shift and are added. Inspired by the current lack of existing emoji embeddings and their limitations to track the evolving emotional content of the emoji, we have created novel emoji embeddings[4] using their emotional contents from their dictionary meanings. Improving emoji embeddings would increase the accuracy of natural language tasks, such as sentiment, emotion, and emoji prediction.

4   Emoji Representation Learning Algorithm

One way to learn high-quality emoji embeddings is by using a logistic regression framework where an input word in the dataset and an output emoji are used to provide a Boolean value to indicate if they are associated or not [7]. This is in stark contrast to training traditional static word embedding models where a target word is related to the context word, through their weighted raw co-occurrence counts. In our dataset, the input is a descriptive word such as "indifferent" and the output is an emoji such as "neutral face" 😐, as well as a label for whether they are associated e.g., True or False. For every dataset sample consisting of an emoji and a descriptive word or phrase, we use Google News word2vec embeddings and calculate the sum of the individual word vectors, v_j represented as:

$$v_j = \sum_{k=1}^{N} w_k \tag{1}$$

where the sequence of words describing that emoji are $w_1, ..., w_n$, which helps model the *approximate* underlying meaning of that emoji. A trainable vector for every emoji in our training set is defined as x_i. The probability of a match between the emoji representation x_i and its description representation v_j is modelled using the dot product, which indicates the similarity between the emoji representation and description vectors. Then to turn the dot product into a probability score, the sigmoid function $\sigma()$ is used. This calculates a prediction and is compared against the target value, which is either 1 or 0, to calculate

---

[4] https://github.com/elenabarry/emojional

how much error is in the model's prediction. This error can be modelled by computing the difference between the target score and the calculated sigmoid score. We then used the logistic loss framework to model this error score to adjust the embeddings so that the next time the model makes this calculation, the result would be closer to the target scores, in an iterative way. Then the model proceeds to the next sample and repeats the process. Our loss function $\mathcal{L}$ is defined as follows:

$$\mathcal{L}(i, j, y_{ij}) = -\log(\sigma(y_{ij}x_i^T v_j - (1 - y_{ij})x_i^T v_j)) \tag{2}$$

where $y_{ij} = 1$ if description j is valid for emoji i and 0 otherwise. However, if all the examples are true, the model would achieve 100%, but may not learn, generating insufficient embeddings. Using negative samples, which are samples that are not associated, is a challenge that the model needs to overcome, creating reliable embeddings in the process. This, in spirit, follows the notion of negative subsampling in [12]. Eisner et al. [7] found that having one positive example per negative example produced the best results in terms of generalized performance. We used a PyTorch model[5], which trains in batches of 8, 4 positive and 4 negative examples, at a learning rate of 0.001. The model performs early-stopping on a held-out development set using 60 epochs of training. Various metrics, including an accuracy and F1 score are outputted.

5   Experiments and Results

In this section, we describe our experiments and compare our model with the baseline models.

5.1   Dataset

We used the latest emoji list, v13.1, from Unicode.org. To create the dataset, we found the best emoji dictionaries that had captured the emotional information were Emojipedia[6], an authoritative voice on emoji usage, and Emoji.Wiki[7], which invites users to add to individual emoji descriptions. Combining descriptions from these websites, and using our knowledge of the emoji, allowed us to collect a full range of emotions for each of the 1816 emojis. By using a list of uniquely emotive, sensory and other keywords we were able to use the Python library Beautiful Soup to scrape any matched words for each emoji description. We created a dataset by crawling online emoji dictionaries for emotions, feelings, sensations and other keywords from emoji descriptions, resulting in 10,854 descriptions of all the current 1816 emojis (v13.1 of Unicode[8]). Figure 1 shows an example of an uncommon emoji.

----

[5] https://github.com/pwiercinski/emoji2vec_pytorch

[6] https://emojipedia.org

[7] https://emojis.wiki

[8] https://unicode.org/emoji/charts/full-emoji-list.html

💉 Person in Suit Levitating

Fig. 1. Example of the collected data for the emoji 💉 : mobster, exclamation mark, business, superiority, prosperity.

We decided not to include skin colour variants, and stick to the original yellow emojis. There are several reasons for this. Firstly, Robertson et al., [15] found that there was no specific sentiment associated with specific skin tone modifiers and that they are used mainly as self-representation. Secondly, this would have exponentially increased editing time, as each emoji has at least five different variants e.g., light, medium-light, medium, medium-dark, dark. Thirdly, the skin tone of an emoji is encoded after the original yellow emoji, this means that all these emojis are still embedded, just their skin tone isn't included. However, this is something that we would like to include in the future. In the meantime, the dataset can be easily adapted by using the keywords light, medium-light, medium, medium-dark, dark similarly to Eisner et al. [7].

After analysing the whole emoji set, we realised that emojis can contain the following information: visual description, underlying nuances such as emotions, feelings and sensations and social or political context. The "raised fist" ✊ emoji illustrates this. However, not all emojis contain all of this information.

1. Visual representation and physical description = **raised fist**
2. Underlying emotional nuances = **resistance, defiance**
3. Social or political contexts = **black lives matter**

This structure informed our dataset samples:

```
✊ raised fist, resistance, defiance, black lives matter
👊 oncoming fist, fist bump, punch, respect, congratulate, fight, greeting
🤛 left-facing fist, fist bump, punch, respect, fight, greeting
🤜 right-facing fist, fist bump, punch, respect, fight, greeting
👏 clapping hands, applause, appreciation, success, achievement, sarcasm
🙌 raising hands, rejoicing, celebrating, success
```

Fig. 2. Dataset Sample

We split our final dataset into train, test and development sets, where our training set consisted of 9,964 true samples, and our test and development sets contained 445 true samples and 445 false samples each. We trained our model using this train, test and development split using the model proposed in Eisner et al., [7]. The model maps emoji symbols into the same space as the 300-dimensional Google News word2vec embeddings. The resulting "emojional" embeddings can be used in addition to 300-dimensional word2vec embeddings in any application.

5.2   Quantitative Results

We used the embedding output of the model and combined it with existing word embeddings, GloVe [13]. We followed the same testing methods as Eisner et al. [7] and used the Random Forests model and Linear SVM for the task of sentiment analysis on the dataset created by Novak et al. [11] in 2015. The model we have proposed achieved the highest accuracy in nearly all tests using the Random Forests algorithm, performing exceptionally well on the 90% most frequent emojis for both algorithms, where our model also outperformed all other embeddings using Linear SVM classification.

The linear SVM classification algorithm creates a dividing line to separate the data into classes, making it susceptible to outliers. Where our dataset has been cleaned the most, there are fewer outliers, and our model outperforms all models, for example, the classification accuracy on the 90% most frequent emojis. However, on less popular emojis our data has been less modified (see Section 5.3) and therefore is more susceptible to outliers.

On the other hand, Random Forests consists of a large number of individual decision trees that operate as an ensemble. It encapsulates the idea of the wisdom of crowds, where each class with the most votes is the model's prediction. This makes it resilient against outliers, allowing our model to outperform other models.

Classification accuracy on entire dataset, N = 12920		
Word Embeddings	Random Forests	Linear SVM
Google News	57.5	58.5
Google News + Barbieri et al [3]	58.2	60.0
Google News + emoji2vec [7]	59.5	60.5
Google News + Sense_des [2]	59.1	62.2
Face Emojis [8]	58.8	**63.3**
Google News + FuzzyMoji2Vec [1]	59.8	61.8
Google News + Emojional	**60.4**	62.3
Classification accuracy on tweets containing emoji, N = 2295		
Google News	46.0	47.1
Google News + Barbieri et al [3]	52.4	57.4
Google News + emoji2vec [7]	54.4	59.2
Google News + Sense_des [2]	50.8	55.1
Face Emojis [8]	58.6	**62.9**
Google News + FuzzyMoji2Vec [1]	**59.2**	62.6
Google News + Emojional	57.0	58.9
Classification accuracy on 90% most frequent emoji, N = 2186		
Google News	47.3	45.1
Google News + Barbieri et al [3]	52.8	56.9
Google News + emoji2vec [7]	55.0	59.5
Google News + Sense_des [2]	50.2	55.3
Face Emojis [8]	59.3	62.1
Google News + FuzzyMoji2Vec [1]	58.7	61.6
Google News + Emojional	**62.0**	**63.0**
Classification accuracy on 10% least frequent emoji, N = 308		
Google News	44.7	43.2
Google News + Barbieri et al [3]	53.9	52.9
Google News + emoji2vec [7]	54.5	55.2
Google News + Sense_des [2]	50.0	50.6
Face Emojis [8]	53.5	54.8
Google News + FuzzyMoji2Vec [1]	54.8	**58.1**
Google News + Emojional	**54.9**	57.5

Table 1. Classification Results

It is also interesting to note the 14.7% (Random Forests) and 17.9% (Linear SVM) difference in accuracy between word embeddings on their own, and word embeddings combined with emoji embeddings, for the classification accuracy on the 90% most frequent emojis. This highlights the need to include emoji embeddings in downstream NLP tasks.

5.3   Qualitative Results

Projecting Emojis onto Emotions, Feelings, Sensations and Keywords: We can naturally match word embeddings to their closest emoji embeddings. This allows us to predict the closest emojis to particular emotions, feelings, sensations or keywords. Our mapping process uses the idea of emojis as synonyms that are matched by their vector proximity. An emoji's pictorial nature means they have a large synonym range. We used a list of Plutchicks emotions such as joy and sadness, types of humour such as sarcasm and irony, feelings such as feeling lucky and emoji categories such as seasonal, celebration and profession types. We used these keywords to map the top 20 emojis to each label. By doing this we could spot outliers and edit their data entries manually, where the emoji did not correlate to the correct category. This allowed the embeddings to be mapped in an intuitive way mimicking the way people use emojis in natural language. In Figure 4 we present an example of using Plutchick's emotions [14] to map the top-5 emojis.

Figure 4 show emojis such as the "Person Bowing" 🙇 predicted for both the categories "admiration" and "submission". This is an example of an emoji that contains multiple emotional senses. This emoji would also be successfully predicted using the words "worship" or "religion".

Figure 4 also demonstrates that inanimate object emojis can have emotional senses. Some emojis predicted in the category "rage" picture inanimate objects, such as the emoji "Axe" 🪓, "Teacup Without Handle" 🍵 and "Boxing Glove" 🥊. As well as their literal sense, they also contain a lateral symbolic sense for rage. This lateral sense comes from the universal symbols that they depict i.e., the Apple version of the "Teacup Without Handle" emoji 🍵 is depicted boiling, a lateral sense for the idea of boiling with rage.

An example of our mapping process is that we would look at anomalies like the "locked" emoji 🔒 in the category "vigilance". Although this emoji projects a universal symbol of safety and trust, which is a close concept to vigilance, using our knowledge of how emojis are used, the "locked" emoji 🔒 is generally not weighted this high in vigilance. Therefore, we would look at the keywords in our dataset for the "locked" emoji 🔒 to see which particular words were giving this emoji such a high vigilance weighting. Usually, the issue was due to duplicate descriptions or close synonyms used which would weight the embedding in a particular direction.

We also present results in the form of t-SNE visualisation in Figure 5 where we can see clusters of emotions in the embeddings in 2D space. This reinforced the idea that specific emotion categories should not be used, as seemingly sim-
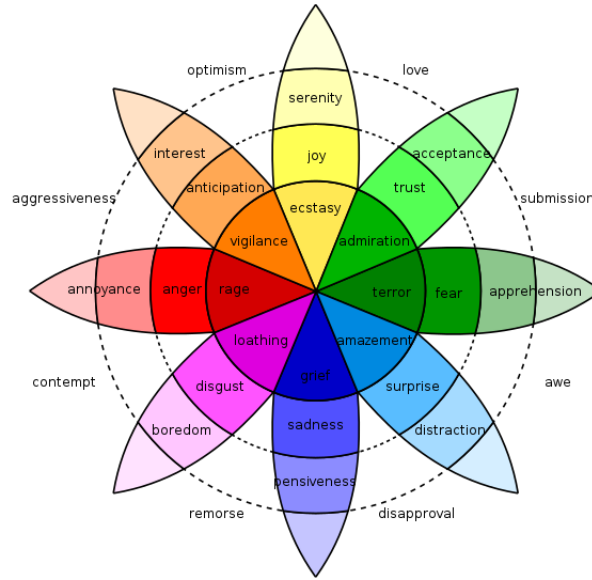
Fig. 3. Pluchik's Wheel of Emotion [14]

ilar emojis are sometimes represented with large distances between each other, showing these emojis are multifaceted when it comes to representing emotion.

6   Conclusion

This paper developed a new approach to emotionally categorize emoji sentiment in novel emoji embeddings. In contrast to other research, each emoji can have a range of emotional content. The changing landscape of emoji makes it time-consuming to create new emoji datasets. The model we have proposed is an emoji embedding that has multifaceted emotions and is not constrained to one emotion classification. We have developed an emoji embedding that outperforms competing models in sentiment analysis when using the Random Forests algorithm. The results present a strong correlation between emoji and emotions, especially with the popular emojis, for which the model scored the highest for both Random Forests and Linear SVM. As these embeddings were also trained on keywords, the subsequent embeddings are durable and can be successfully used in different natural language tasks such as emotion, cyberbully and sarcasm detection. For future development, it would be interesting to use this model for emoji prediction, as these embeddings create emoji synonyms for words that could be substituted or combined with text suggestions. This would mirror the intuitive summative process that people use when selecting emojis for their text.

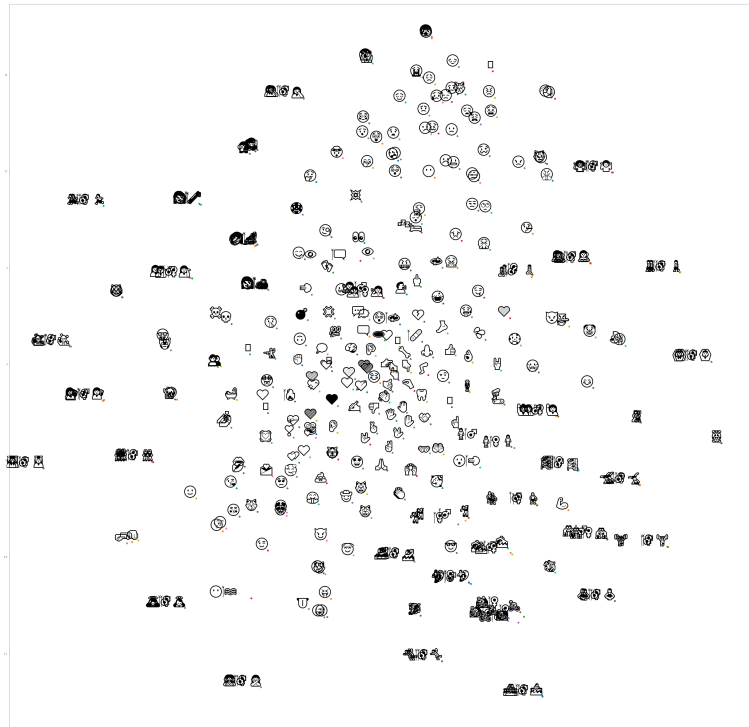**Fig. 4.** Emoji Emotion Prediction

Fig. 5. t-SNE Popular Emojis

References

1. Ahanin, Z., Ismail, M.A.: Feature extraction based on fuzzy clustering and emoji embeddings for emotion classification. IJTMIS 2(1), 102–112 (2020)
2. Ai, W., Lu, X., Liu, X., Wang, N., Huang, G., Mei, Q.: Untangling emoji popularity through semantic embeddings. In: AAAI. vol. 11 (2017)
3. Barbieri, F., Ronzano, F., Saggion, H.: What does this emoji mean? a vector space skip-gram model for twitter emojis. In: ELRA p. 3967-72. ELRA (European Language Resources Association) (2016)
4. Chen, Y., Yuan, J., You, Q., Luo, J.: Twitter sentiment analysis via bi-sense emoji embedding and attention-based lstm. In: ICM. pp. 117–125 (2018)
5. Derrac, J., Schockaert, S.: Inducing semantic relations from conceptual spaces: a data-driven approach to plausible reasoning. AIJ 228, 66–94 (2015)
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL. pp. 4171–4186 (2019)
7. Eisner, B., Rocktäschel, T., Augenstein, I., Bošnjak, M., Riedel, S.: emoji2vec: Learning emoji representations from their description. In: Workshop on NLP for Social Media. pp. 48–54. Association for Computational Linguistics, Austin, TX, USA (Nov 2016), https://www.aclweb.org/anthology/W16-6208
8. Guibon, G., Ochs, M., Bellot, P.: From emoji usage to categorical emoji prediction. In: CiCLING (2018)
9. Harris, Z.S.: Distributional structure. Word 10(2-3), 146–162 (1954)
10. Hill, F., Cho, K., Korhonen, A., Bengio, Y.: Learning to understand phrases by embedding the dictionary. TACL 4, 17–30 (2016)
11. Kralj Novak, P., Smailović, J., Sluban, B., Mozetič, I.: Sentiment of emojis. PloS one 10(12), e0144296 (2015)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
13. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: EMNLP. pp. 1532–1543 (2014)
14. Plutchik, R.: A psychoevolutionary theory of emotions (1982)
15. Robertson, A., Magdy, W., Goldwater, S.: Self-representation on twitter using emoji skin color modifiers. In: AAAI. vol. 12 (2018)
16. Yang, M.S.: A survey of fuzzy clustering. Mathematical and Computer modelling 18(11), 1–16 (1993)
17. Zhou, J., Zogan, H., Yang, S., Jameel, S., Xu, G., Chen, F.: Detecting community depression dynamics due to covid-19 pandemic in australia. IEEE TCSS (2021)