# RNN for Affects at SemEval-2018 Task 1: Formulating Affect Identification as a Binary Classification Problem

**Aysu Ezen-Can**
SAS Inst.
aysu.e.can@gmail.com

**Ethem F. Can**
SAS Inst.
ethfcan@gmail.com

## Abstract

Written communication lacks the multimodal features such as posture, gesture and gaze that make it easy to model affective states. Especially in social media such as Twitter, due to the space constraints, the sources of information that can be mined are even more limited due to character limitations. These limitations constitute a challenge for understanding short social media posts.

In this paper, we present an approach that utilizes multiple binary classifiers that represent different affective categories to model Twitter posts (e.g., tweets). We train domain-independent recurrent neural network models without any outside information such as affect lexicons. We then use these domain-independent binary ranking models to evaluate the applicability of such deep learning models on the affect identification task. This approach allows different model architectures and parameter settings for each affect category instead of building one single multi-label classifier. The contributions of this paper are two-folds: we show that modeling tweets with a small training set is possible with the use of RNNs and we also prove that formulating affect identification as a binary classification task is highly effective.

## 1 Introduction

Social media platforms allow users to share information, communicate with other users, learn about new products, and get latest news. The importance of social media data is getting larger every day as social media usage grows every year (Duggan, 2015). Twitter is one such social media platform where users can write short posts as well as share links. Twitter is also used for getting news (Center, 2017).

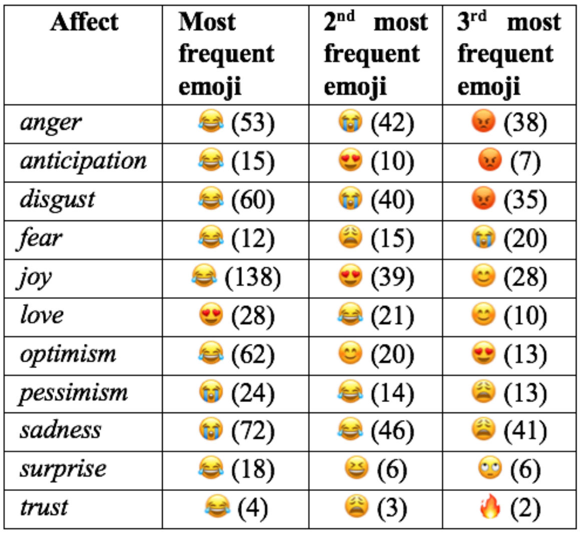
A large body of research has been conducted using Twitter data including analyzing user intentions (Java et al., 2007), determining influence of users (Romero et al., 2011), predicting retweet counts (Can et al., 2013), classifying sentiments of tweets (Jansen et al., 2009; Agarwal et al., 2011; Neethu and Rajasree, 2013; Kontopoulos et al., 2013; Pak and Paroubek, 2010). All of these studies have one goal in common: understanding/modeling information diffuse in Twitter.

One aspect of modeling social media posts is focusing on emotional states of users. There has been plenty of efforts on determining affective states (Schwarz and Clore, 1983) and their effects to human behavior for different domains from education (Sidney et al., 2005) to health care (Lisetti et al., 2003). For Twitter, this problem is even more challenging as the information source is limited to the number of characters allowed in a single post and multimodal features (e.g., posture, gesture, and eye gaze) are not available.

In this paper, we formulate affect identification task as a binary classification problem and investigate the applicability and effectiveness of domain-independent deep learning models as well as features. Our dataset includes eleven affect categories (i.e., anger, anticipation, disgust, fear, joy, love, optimism, pessimism, sadness, surprise, and trust) for each tweet. The presence of one affect category in a tweet does not stop another category to be present (e.g., joy and optimisim can both be present in a tweet). We represent each affect category as one class and build binary classifiers for each class. Recurrent neural networks are trained for each affect category and no domain-dependent features such as affect lexicons are used. Our goal is to evaluate a generic model for different affective states.

Binary models have been successfully applied to several applications including action recognition in videos (Can and Manmatha, 2013), prediction of whether or not a tweet will be

*Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018)*, pages 162–166
New Orleans, Louisiana, June 5–6, 2018. ©2018 Association for Computational Linguistics



| Affect | Most frequent emoji | 2nd most frequent emoji | 3rd most frequent emoji |
|---|---|---|---|
| *anger* | 😂 (53) | 😭 (42) | 😡 (38) |
| *anticipation* | 😂 (15) | 😍 (10) | 😡 (7) |
| *disgust* | 😂 (60) | 😭 (40) | 😡 (35) |
| *fear* | 😂 (12) | 😩 (15) | 😭 (20) |
| *joy* | 😂 (138) | 😍 (39) | 😊 (28) |
| *love* | 😍 (28) | 😂 (21) | 😊 (10) |
| *optimism* | 😂 (62) | 😊 (20) | 😍 (13) |
| *pessimism* | 😭 (24) | 😂 (14) | 😩 (13) |
| *sadness* | 😭 (72) | 😂 (46) | 😩 (41) |
| *surprise* | 😂 (18) | 😆 (6) | 🙄 (6) |
| *trust* | 😂 (4) | 😩 (3) | 🔥 (2) |

Figure 1: Most frequently used emojis and their counts for each affect category.

retweeted (Hong et al., 2011), and topic classification (Joachims, 1998). In this paper, we describe our approach for affect recognition of English tweets (Task E-c: Detecting Emotions), a subcategory of Task 1 in the SemEval 2018 challenge (Mohammad et al., 2018).

## 2 Corpus

In this paper, we use English tweets that have been annotated by affect categories (Mohammad et al., 2018). The dataset contains emojis, hashtags, and the textual content of tweets; however, it does not have user ids. The training, validation, and test splits are done by the task organizers. Figure 1 shows top three mostly used emojis in each class and their frequencies for the training set.

### 2.1 Breakdown of Emojis to Classes

Due to the importance of visual cues in predicting affective states, we pay attention to a form of visual cues: emojis. Here we present some of our findings based on different affect categories.

- *Trust*: emojis are not frequently used. Not easy to determine through emojis.

- *Sadness*: The sobbing face emoji is expectedly the most common one but interestingly laughing with joy emoji is the second most common. Weary face emoji is also very common in sadness: 56.16% of all weary face emojis are used in this class.

- *Anger* and *disgust* share the same property: the most common emoji is the laughing with joy emoji and the second most common is sobbing face emoji. The fact that a joy emoji being the most commonly used in these affective classes is quite interesting and can indicate irony. The third most common emoji in these two classes are also the same: rage emoji.

- An emoji that can be intuitively associated with love (heart eyes) actually occurs more in *joy* tweets than *love* tweets.

- An unexpected finding is on fire emoji where *joy* and *optimism* classes have a large portion of all fire emojis in the training set (46.7% and 36.7% respectively).

- The affective class that uses most emojis is *joy*.

## 3 Methodology

Since each tweet in the data contains eleven affect categories (i.e., anger, anticipation, disgust, fear, joy, love, optimism, pessimism, sadness, surprise, and trust), we created eleven datasets with the same tweets but with different class information. For example, the first dataset has one values (i.e., positive) for tweets that show anger and zero (i.e., negative) for those that do not have anger. Other datasets are created in the same way for the remaining affects classes. By building one model for each affect category, we formulated affect identification problem as a binary classification task. Then in testing time, we obtained predictions from every specific model and fused the results to obtain a unique result for each tweet.

### 3.1 Training Binary Classification Models

The advantage of using binary classification models for each affect category is that each model can be trained by itself, enabling different model architectures and parameters. For example, while one category may benefit from a deeper model, the other affect category can obtain the best results with a shallower model. In this way, the models do not have to be the same for each affect class.

### 3.2 Model Architecture

We built separate RNN models for each affect category, resulting with eleven classifiers. For the classifiers, we used three GRU layers, two of

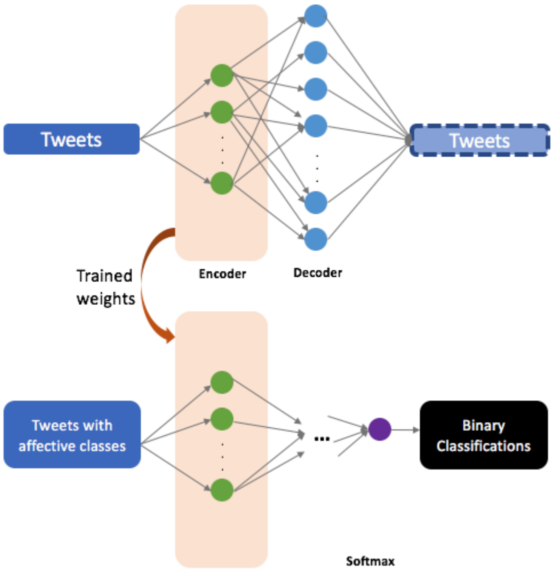


Figure 2: Using the unlabeled tweets for training an auto-encoder and using the trained weights for the affect classification process.

which are bi-directional. To be able to build a more generalized model, a dropout of 0.2 is used in each layer. Each bidirectional layer contains 100 neurons and the final encoding layer has 50 neurons.

## 3.3 Training Auto-Encoder

Because the dataset is not very big, we wanted the classifiers to learn as much information as possible without overfitting it. Therefore, we built an auto-encoder from the tweets' content (e.g., unlabeled tweets, no affect categories). The goal of the auto-encoder is to get weights that can be used in the classifiers. As shown in Figure 2, we used the trained weights from the auto-encoder to start building binary classifiers. To convert a text-generating auto-encoder into a classifier, we added a softmax layer.

## 3.4 Features

For modeling affect categories in tweets, we use only the words and emojis. No domain-dependent features, or features that are aware of task in hand (e.g., affect lexicons) are used as our goal is to determine how well a generic RNN model can perform for affect recognition task.

### 3.4.1 Emojis

To represent emojis as embeddings, we used the pre-trained embeddings from the emoji2vec package (Eisner et al., 2016).

### 3.4.2 Word Embeddings

For this study, an embedding length of 200 is used. We utilized pre-trained global vectors trained on tweets (Pennington et al., 2014)

### 3.4.3 Hashtags

Hashtags have a lot of semantic information about the tweets. However, most of that information is neglected if the hashtags cannot be found in the words embeddings. Therefore, we followed a greedy approach for dividing hashtags into their corresponding words.

Once the # is removed, we take the content of the hashtag and search if the content is present in the vocabulary as its entirety. If vocabulary has the hastag content, we use it. If not, more processing is done. Starting from the beggining of the word we keep a pointer, searching for a valid word that from index=0 to index=pointer. Once 0,j indices represent a substring that is a valid word, we continue the recursive search for the rest of the content (i.e., j+1 to the end of the string). The words that are found are added to the list of words that represent the hashtag. Then we use those words and represent them as embeddings.

Because this approach is greedily finding the shortest possible words contained within the hashtag, it is not guaranteed to represent the correct semantics all the time. For example, the #feelsadforyou is correctly divided to ['feel',"sad', 'for', 'you'], however, #toniteinasheville ('tonite in asheville") becomes ['tonite', 'in', 'as', 'he', 'ville'], which is not correct. Achieving perfect semantics would require human labeling, therefore, we used the greedy approach and have observed that utilizing hash tag contents significantly improves the effectiveness of the models.

## 3.5 Results on Validation Set

The accuracies of binary classification models for each affect category are presented in Figure 3. We compare the models' performances with majority baselines where the percentage of the class value that occurs most is taken as the majority baseline for each class.

# 4 Results

In this section, we report the results for the test set as well as discussion on the results.

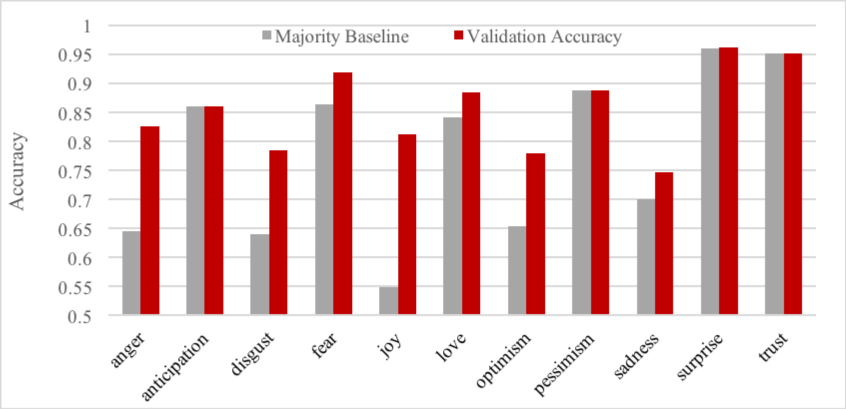


Figure 3: Accuracies per affect category. Majority baseline of each class is compared to the performance of the RNN classifier for that class.

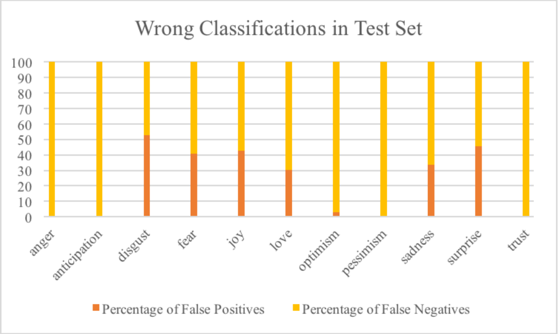


Figure 4: The distribution of false positives and false negatives for observations that are classified incorrectly in the test set.

### 4.1 Experimental Results

For all of our experiments, we used SAS Deep Learning Toolkit. We utilized an environment with 4 workers, with 24 threads in each worker, and mini-batch size per thread on each worker was 6. Adam optimizer is used in all experiments.

Using the test set, the proposed model achieved a 0.398 accuracy, 0.539 micro-avg F1, and 0.358 macro-avg F1. A random baseline achieves 0.185 accuracy, 0.307 micro-avg F1, and 0.285 macro-avg F1. Compared to the random baseline, the generic RNN model is quite successful at identifying affect categories.

### 4.2 Discussion

Some of the affect categories have very few positive examples, therefore it is very difficult for classifiers to learn nuances of those affects. For example, surprise and trust categories have 96.05% and 95.15% majority baselines respectively. In other words, only 4-5% of all training set observations have these affect categories as true.

As can be seen in Figure 4, when the number of positive observations are limited, the classifiers tend to make more false negatives. For affect categories that have a major class value that is dominant, we experimented with sampling as well where the number of positive and negative examples were equal. However, that made the dataset significantly smaller, further making it difficult for the RNN models to learn distinctions. Rather than using smaller datasets or including external data, we prefer to employ binary models. One of the main advantages of using binary models over multi-label models is to better deal with the uneven distribution of positive examples across classes.

## 5 Conclusion

Affect identification without visual cues is a challenging task, making the text as the only source of information that can be used for machine learning models. This problem gets more challenging as the text data gets limited by the number of characters in Twitter.

This paper presented a simple yet effective approach for classifying affect categories of Tweets. The main motivation of this paper was to evaluate how well a domain-independent RNN model can perform for classifying affects. Therefore, no domain-dependent source of information such as affective lexicons or pre-trained affect features are used. We built binary classification models per each affect category. The results showed that RNNs are powerful enough to outperform the baselines significantly, even without prior knowledge about the domain and with a relatively small dataset.

## References

Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment analysis of twitter data. In *Proceedings of the workshop on languages in social media*, pages 30–38. Association for Computational Linguistics.

Ethem F Can and R Manmatha. 2013. Formulating action recognition as a ranking problem. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pages 251–256. IEEE.

Ethem F. Can, Hüseyin Oktay, and R. Manmatha. 2013. Predicting retweet count using visual cues. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management*, CIKM '13, pages 1481–1484, New York, NY, USA. ACM.

Pew Research Center. 2017. News use across social media platforms 2017.

Maeve Duggan. 2015. Mobile messaging and social media. Pew Research Center.

Ben Eisner, Tim Rocktäschel, Isabelle Augenstein, Matko Bošnjak, and Sebastian Riedel. 2016. emoji2vec: Learning emoji representations from their description. *arXiv preprint arXiv:1609.08359*.

Liangjie Hong, Ovidiu Dan, and Brian D. Davison. 2011. Predicting popular messages in twitter. In *Proceedings of the 20th International Conference Companion on World Wide Web*, WWW '11, pages 57–58, New York, NY, USA. ACM.

Bernard J Jansen, Mimi Zhang, Kate Sobel, and Abdur Chowdury. 2009. Twitter power: Tweets as electronic word of mouth. *Journal of the Association for Information Science and Technology*, 60(11):2169–2188.

Akshay Java, Xiaodan Song, Tim Finin, and Belle Tseng. 2007. Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65. ACM.

Thorsten Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer.

Efstratios Kontopoulos, Christos Berberidis, Theologos Dergiades, and Nick Bassiliades. 2013. Ontology-based sentiment analysis of twitter posts. *Expert systems with applications*, 40(10):4065–4074.

Christina Lisetti, Fatma Nasoz, Cynthia LeRouge, Onur Ozyer, and Kaye Alvarez. 2003. Developing multimodal intelligent affective interfaces for telehome health care. *International Journal of Human-Computer Studies*, 59(1-2):245–255.

Saif M. Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 Task 1: Affect in tweets. In *Proceedings of International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA.

MS Neethu and R Rajasree. 2013. Sentiment analysis in twitter using machine learning techniques. In *Computing, Communications and Networking Technologies (ICCCNT), 2013 Fourth International Conference on*, pages 1–5. IEEE.

Alexander Pak and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, volume 10.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Daniel M Romero, Wojciech Galuba, Sitaram Asur, and Bernardo A Huberman. 2011. Influence and passivity in social media. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 18–33. Springer.

Norbert Schwarz and Gerald L Clore. 1983. Mood, misattribution, and judgments of well-being: Informative and directive functions of affective states. *Journal of personality and social psychology*, 45(3):513.

K Dmello Sidney, Scotty D Craig, Barry Gholson, Stan Franklin, Rosalind Picard, and Arthur C Graesser. 2005. Integrating affect sensors in an intelligent tutoring system. In *Affective Interactions: The Computer in the Affective Loop Workshop at*, pages 7–13.