IIT Delhi at SemEval-2018 Task 1 : Emotion Intensity Prediction

Bhaskar Kotakonda
Undegraduate
IIT Delhi
ee1140448@iitd.ac.in

Prashanth Gowda
Undergraduate
IIT Delhi
ee1140470@iitd.ac.in

Brejesh Lall
Associate Professor
IIT Delhi
brejesh@ee.iitd.ac.in

Abstract

This paper discusses the experiments performed for predicting the emotion intensity in tweets using a generalized supervised learning approach. We extract 3 kind of features from each of the tweets - one denoting the sentiment and emotion metrics obtained from different sentiment lexicons, one denoting the semantic representation of the word using dense representations like Glove, Word2vec and finally the syntactic information through POS N-grams, Word clusters, etc. We provide a comparative analysis of the significance of each of these features individually and in combination tested over standard regressors available in scikit-learn. We apply an ensemble of these models to choose the best combination over cross validation.

Our resources and the details of implementation are publicly available at :

`https://github.com/prashanth470/affect-in-tweets`

1 Introduction

In Natural Language Understanding, the field of sentiment analysis deals with the process of determining the polarity of a given text, such as positive, negative, neutral and mixed. In extension to this analysis, we have the emotion recognition task which deals with associating the text with predefined set of emotions like anger, fear, joy, etc. A general method of performing the emotion recognition task is to employ weak supervision models like emojis, hashtags and emoticons to mine emotion. Instead of using this discreet approach to emotion, continuous models that map text to an n - dimensional space with valence, arousal and dominance can be used.

Another interesting problem in the NLP space is the abundance of social media texts, especially twitter. Twitter is a micro-blogging site where people express themselves and react to content in real-time. An estimated 500 million tweets are generated on a daily basis. The peculiar nature of such micro-blogging sites is the form of expression through hashtags, emojis, slang and informal words etc. But analyzing this abundant information would help us to realize several insights about an event, person or organization.

It is with this motivation that the SemEval shared task on Emotion Intensity was conducted.(Mohammad et al., 2018) Given a tweet and an emotion (anger, fear, sadness, joy) the aim is to determine the intensity score that can be seen as an approximation to the intensity felt by the reader or expressed by the author. The paper is divided into 3 sections hereon - the second section talks about the system description, the third section on a comparative analysis of results and finally a discussion on the future scope.

2 System Description

The datasets for anger, joy, fear and sadness were created using a technique called the Best Worst Scaling.(Mohammad and Kiritchenko, 2016) These annotations lead to reliable fine grained intensity scores which can be used to imply the intensity or the degree of an emotion expressed. The detailed data collection information can be found in Mohammad et al.(Mohammad and Turney)

2.1 Pre Processing

This step includes modifying the raw tweets to a form that can be easier to process for the further steps. It has already been asserted that the nature of the text in question is peculiar as it is mined from social media. In addition to regular usage of words emoticons, user ids and URLs are common in social media. It is very important to note that

Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018), pages 339–344
New Orleans, Louisiana, June 5–6, 2018.

while the tweet is tokenized into words, the process is twitter-aware, or the splitting is done keeping in mind the utility of User IDs and URLs as separate entities.

We tried 2 kinds of tokenizers : tweetokenize and regular expressions using the regex expression matching in python. We demonstrate below the difference in tokenizing for each of these and why we chose tweetokenize as it was more tweet aware.

The sentence used is :
What are some good #funny #entertaining #interesting accounts I should follow ? My twitter is dry

1. **Regex Python**
   'what', 'are', 'some', 'good', 'funny', 'entertaining', 'interesting', 'accounts', 'i', 'should', 'follow', '?', 'my', 'twitter', 'is', 'dry'

2. **Tweetokenize**
   'What', 'are', 'some', 'good', '#funny', '#entertaining', '#interesting', 'accounts', 'I', 'should', 'follow', '?', 'My', 'twitter', 'is', 'dry'

2.2 Feature Extraction

The baseline feature made available is the **Affective Tweets** package, which includes a number of lexicon based and syntactic feature extraction modules. After a thorough analysis of various systems of NLP competitions from Kaggle, KD Nuggets and various other conferences, we narrowed down to 3 type of important features.

1. **Lexicon Based**
   Many of the tasks related to sentiment and emotion are using these features extensively (Mohammad and Kiritchenko, 2018). A lexicon is a dictionary of words with labels specifying their sentiments and scores to identify the intensity of text. Table 1 shows the different lexicons used, the scores they contribute and the size of the corpus. Using the above features selectively leads to a 135 dimensional feature vector, which as we observe is still relatively sparse with only a few non zero values.

2. **Semantic Based**
   To overcome limitations of using the sparse lexicon based features and to add the semantic meaning of the words, compactly represented low dimesional dense vector encodings called word embeddings are also included. Glove embeddings, which are 200 dimensional vectors trained on 2 billion corpus are integrated. Although these vectors accurately represent the significance of a word in a context, the sentence embeddings or the representation in a sequential manner is not focused on in this section. The sentence embedding is considered to be the average of the individual word embeddings of the sentence. The final represented sentence embedding is a 25 dimensional vector.

3. **Syntactic Based**
   Although the meaning of the individual words have been taken into account in the semantic based vectors, it is essential to encode certain other aspects of the word, like part of speech tags, brown clusters and word n grams.

The final feature vector is chosen based on the significance of each of the individual features, when input to regressors to maximize the pearson coefficient.

2.3 Regressors

Each of the above features have very little correlation between each other as they represent different aspects of the text. Hence the regressors such as Support Vectors Regression, AdaBoost, Random Forest Regressor and Bagging regressor, etc can be used effectively. The feature vectors are used without any kind of normalization.

2.4 Hyper Tuning

The sci-kit package enables an extensive grid search mechanism to find the optimum value of the various hyper parameters of a regressor. Figure 1 shows the different values of C and gamma taken by the regressor to maximize the cost function of pearson coefficient using 10 fold cross validation. It shows anger, fear, joy and sadness metrics in a clockwise manner. Table 2 also shows the parameter values of the SVR for different emotions.

The best combination of hyper parameters are denoted by the grey spot in the grid search for each of the emotions.

Affect Lexicon	Description	Corpus Details
NRC Hashtag Emotion	Positive and negative variables by aggregating the positive and negative word scores provided by this lexicon created with tweet annotated with emotional hashtags	**emotions**: anger, anticipation fear, joy, sadness, surprise, trust
		size : 16,862 unigrams
		score : 0 to infinite
Sentiment140	Aggregating positive and negative scores	**emotions** : anger, fear
		size : 45,255 unigrams
		score : -inf to +inf
NRC 10	Adds the emotion associations of the words matching the Twitter Specific expansion	**emotions** : +ve, -ve
		size : 62,468 unigrams
		score : -inf to +inf
SentiStrength	Weighted average of the sentiment distributions of the synsets for word occurring in multiple synsets	**emotions** : anger, anticipation, fear, joy, sadness, surprise, trust
		size : 14,000 words
		score : 0 to 1
NRC Emotion	Calculates a positive and a negative score by aggregating the word associations provided by a list of emoticons	**size** : 76,400 terms
		score : count

Table 1: Lexicons used for feature extraction.

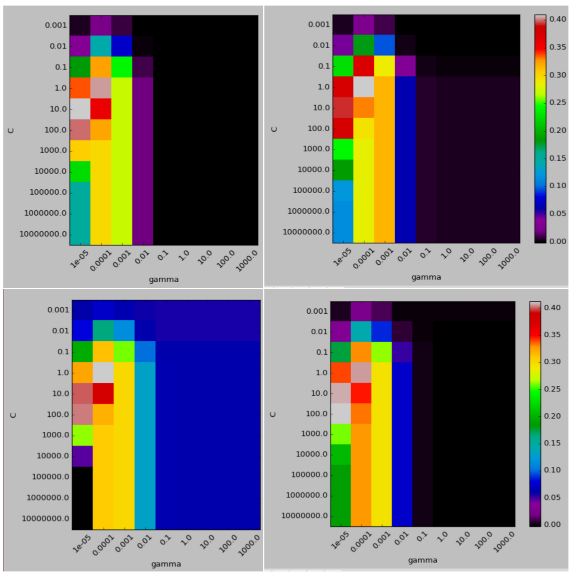

Figure 1: Hypertuning of SVR.

Emotion	C	Gamma
Anger	100	1e-05
Fear	1.0	1e-04
Joy	1.0	1e-04
Sadness	10	1e-05

Table 2: Final Parameters for Support Vector System.

3 Results

Only the semantic and lexicon based features are seen to be having a positive affect on the pearson coefficient while the syntactic feaatures show almost no improvement. Hence, they are discarded from further analysis. The 10 fold cross validation shows best performance in the case of employing all the different lexicons available in concatenation with the average word embedding.

3.1 Experimental Results

Table 4 shows the performance of this feature vector when trained across various regressors. The gradient boosting with XGBOOST ensemble regressor is observed to give the best results. The spearman coefficient has been skipped in the analysis as it had the same insights offered by the pearson coefficient.

Emotion	Tweet	Predicted	Gold
Anger	'Pope fuming after police broke up drug-fuelled Vatican priest gay orgy' some headline.	0.539	0.545
	Lugubrious face, crestfallen eyes, forlorn heart and an agitated soul seeking serenity.	0.418	0.437
	I talked to an Asian yesterday.	0.374	0.000
	You are MINE, my baby, my headache, my love, my smile, my frown, my wrong, my right, my pain, my happiness, my everything.	0.433	0.172
Fear	i'm nervous	0.777	0.741
	The moment I joined BTS, I was nervous and amp; felt lost. I still have those feelings but whenever I do, the people who bring me back are you guys	0.713	0.732
	Considering I am 101% fine with getting tattoos, blood tests terrify me and I AM HAVING TO GET ONE AAAAAHHHH	0.435	0.845
	Every time I fart my dog jumps in fear hahahaha lol	0.610	0.242
Joy	Streetlights coming on. We can see stars! #amazing #SolarEclipse2017	0.681	0.672
	@SteveConteNYC lovely! :)	0.661	0.672
	My dads big day is only less than 2 weeks away!	0.456	0.844
	What do you call a camel with no humps? Humphrey! #joke #writerslife #WednesdayWisdom	0.286	0.6
Sadness	Do not linger too long near the howff or you risk the displeasure of a chuhaister with pubes more underwhelming than those of an aurochs.	0.427	0.417
	@rohandes Lets see how this goes. We falter in SL and this goes downhill. :(	0.377	0.385
	I wonder how many Lexas and Alexandrias there will be in 10 tears.	0.653	0.321
	Me at Start of Semester Expecting = A+ After Mids = B+ After Finals = Passing Marks. Thinking to quit	0.343	0.696

Table 3: Analysis of sample predictions in each emotion.

3.2 Limitations

The features that were chosen to represent the sentences, although having limitations in terms of missing context, perform significantly well in estimating the emotion. Table 3 analyzes the system's predictions in cases were the gold labels were close to the final value as well as the erroneous cases.

In cases where there are multiple instances of displaying emotion the model is very successful as seen in the first samples of every emotion. We also observe that the emoticons and punctuation are very well accounted for, like *@SteveConteNYC lovely! :)* and *@rohandes Lets see how this goes. We falter in SL and this goes downhill. :(*. It can also be said that the model is twitter aware as it often attributes an intensity based on the relative importance of the hashtag and emoticon.

There are broadly three cases where the system has trouble - one where there is very little context to decide an emotion, which is problematic even for the manual annotation, like *I talked to an Asian yesterday*. This should not be misunderstood with racial bias but merely a lack of training data. The second case is Sarcasm, like *Every time I fart my dog jumps in fear hahahaha lol*. While the lexicon based features attribute high intensity of fear due to direct usage of the word *fear*, it has to be understood that words such as *hahahaha, lol* have a diminishing effect on this sentiment. Finally, we

Regressor	Emotion	Pearson	Pearson (> 0.5)	Spearman (> 0.5)
SVR rbf (gamma=0.001) (C=1e-4)	Anger	0.607	0.349	0.346
	Fear	0.627	0.441	0.412
	Joy	0.415	0.328	0.331
	Sadness	**0.622**	0.483	0.493
Random forest	Anger	0.614	0.501	0.505
	Fear	0.556	0.411	0.389
	Joy	0.569	0.359	0.358
	Sadness	0.541	0.435	0.429
Adaboost	Anger	0.585	0.385	0.389
	Fear	0.612	0.502	0.483
	Joy	**0.638**	0.408	0.433
	Sadness	0.579	0.423	0.437
Gradient Boosting	Anger	**0.660**	0.506	0.516
	Fear	**0.677**	0.500	0.480
	Joy	0.622	0.380	0.393
	Sadness	0.606	0.497	0.506
Bagging	Anger	0.576	0.421	0.425
	Fear	0.590	0.459	0.429
	Joy	0.546	0.353	0.337
	Sadness	0.570	0.444	0.446

Table 4: Results over different regressors.

also see that in cases where there is no direct usage of the words from lexicon but merely the context of the preceding sentences that decide the emotion, like *Me at Start of Semester Expecting = A+ After Mids = B+ After Finals = Passing Marks. Thinking to quit MS.* This is quite expected due to the choice of features we employed.

4 Future Work

The main limitation of this approach is overlooking the importance of context and compositionality of the sentence, in addition to the semantic and syntactic attributes. This can be taken into account by using bi-directional LSTMs - long short term memory approach. LSTMs allow for learning sentence representations that account for context to be stored in memory over a longer distance through a mechanism of forgetting and memory at each stage, thus tackling the problem of vanishing gradient.(Olah)

Although Convolution Neural Networks have been discovered for image recognition tasks, recent research of (Kim, 2014) show exceptionally high accuracy of CNNs when trained on word embedding for language understanding tasks. The CNNs effectively appply filters of different sizes to images which can be understood as considering a n-gram featurizer and deciding on the most effective n-gram that contributes to the meaning of the tweet.

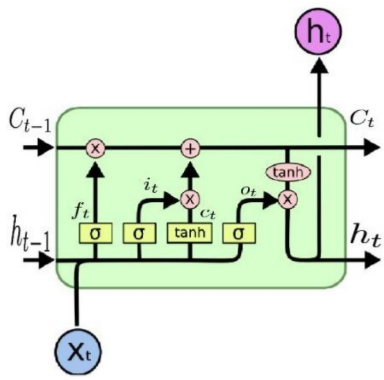

Figure 2: LSTM Model.

Acknowledgments

We thank our supervisor Dr.Brejesh Lall for guiding us through the process and the organizers of SemEval 2018 for providing us the opportunity to work on this task and creating datasets.

References

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882.

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.

Saif M. Mohammad and Svetlana Kiritchenko. 2016. Capturing reliable fine-grained sentiment associations by crowdsourcing and bestworst scaling.

Saif M. Mohammad and Svetlana Kiritchenko. 2018. Understanding emotions: A dataset of tweets to study interactions between affect categories. In *Proceedings of the 11th Edition of the Language Resources and Evaluation Conference*, Miyazaki, Japan.

Saif M. Mohammad and Peter D. Turney. Crowdsourcing a word-emotion association lexicon. New Orleans, LA, USA.

Christopher Olah. Understanding lstm networks.