TELKOMNIKA, Vol.16, No.3, June 2018, pp. 1394~1401
ISSN: 1693-6930, accredited **A** by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v16i3.8387

Text to Emotion Extraction Using Supervised Machine Learning Techniques

Muhammad Anwarul Azim*, Mahmudul Hasan Bhuiyan
Department of Computer Science and Engineering, University of Chittagong,
Chittagong-4331, Bangladesh. Phone: +88031726311
*Corresponding author, e-mail: e-mail: azim@cu.ac.bd

Abstract

Proliferation of internet and social media has greatly increased the popularity of text communication. People convey their sentiment and emotion through text which promotes lively communication. Consequently, a tremendous amount of emotional text is generated on different social media and blogs in every moment. This has raised the necessity of automated tool for emotion mining from text. There are various rule based approaches of emotion extraction form text based on emotion intensity lexicon. However, creating emotion intensity lexicon is a time consuming and tedious process. Moreover, there is no hard and fast rule for assigning emotion intensity to words. To solve these difficulties, we propose a machine learning based approach of emotion extraction from text which relies on annotated example rather emotion intensity lexicon. We investigated Multinomial Naïve Bayesian (MNB) Classifier, Artificial Neural Network (ANN) and Support Vector Machine (SVM) for mining emotion from text. In our setup, SVM outperformed other classifiers with promising accuracy.

Keywords*: affective computing, text to emotion, emotion extraction, machine learning, twitter data analysis*

1. Introduction

Emotion plays a very important role in communication as hundred of words can be told using an emotional expression. The internal state of human is expressed through emotion which is communicated in various forms such as verbal words, facial expressions as well as written text [1]. Expansion of internet has made text communication popular with people of all ages. They are involved with different social media and blogs as those are great means of communication. They express their sentiments and opinions through messages, status, comments, reviews etc. A communication containing emotional content is livelier and also has a great impact on the mind of participants [1]. Therefore, understanding the emotion conveyed through text is very important to ensure proper communication and avoid misunderstanding. Additionally, text to emotion extraction has significant applications in different fields of Human Computer Interaction (HCI) including robotics.

Enormous amount of text are produced daily on different social media and blogs. Manually analyzing the emotion conveyed in these texts is near to impossible and hence automated tools are necessary for the task. The interdisciplinary field of computing that spans computer science, psychology and cognitive science dealing with the systems related to emotions is known as Affective Computing [2-4]. Affective computing researchers have proposed various rule based approaches to extract the emotion of text automatically. Neviarouskaya et al. [3] developed a lexical rule based approach known as Affect Analysis Model (AAM) based on lexicon built from WordNet-Affect [5] where each word in the database is represented by a vector of intensities of nine emotion defined by Izard [6]. They analyzed a sentence in word, phrase and sentence level then provided the emotion with maximum total intensity as output. L. Dey et al. [7] provided implementation details of real time chatting application that showed the emotion of the message by analyzing affective features of the words of the text. They used a modified version of AAM with their own lexicon. An active learning based system which took experts suggestion to continuously train the system and improve the accuracy along with text to emotion extraction was proposed by M.-U. Asad et al. [1]. The Naïve Bayesian classifier was used for active learning phase in that system. The authors further

Received December 19, 2017; Revised April 13, 2018; Accepted May 16, 2018

augmented the exploratory data warehouse (DW) technology to their previous existing system for enhancing their dataset [8]. All these systems relied on emotion intensity lexicon where each word is assigned with a vector of intensity values of different emotions. However, creating emotion intensity lexicon is very time consuming and tedious job. Moreover, there is no hard and fast rule for assigning emotion intensity values to a word. Thus, intensity values vary from person to person which introduces bias in the output. A Machine Learning based system would solve the problems regarding the rule based approcah.

Machine Learning is an approach which is used to enable machine to learn from empirical data validated by experts [9]. Machine learning task is divided into supervised learning and unsupervised learning. In supervised learning, also known as classification, data tuples and their corresponding classes are given to the system and a classification model is developed on the train data. The system predicts the class level of new data based on the classification model. On the other hand, in unsupervised learning known as clustering, no class level is given and system forms different clusters from the data based on any similarity measure. The task of emotion extraction from text can be represented as supervised machine learning problem.

Supervised Machine learning based approaches for sentiment analysis are investigated by different researchers. Those works are confined to finding positivity, negativity and neutral nature of text. Pang et al. proposed machine learning based approach for finding only positivity and negativity in movie review domain [10]. They used the standard Bag of Features framework with Naïve Bayesian classifier, Maximum Entropy and Support Vector Machines. Twitter [11] is popular social media all over the world and great source of emotional data. People express their status through short text form known as Tweets. Sentiment analysis of tweets on GSM services is experimented in [12] where the authors used Multinomial Naïve Bayes for classification. A methodology proposed in [13] is related to our work which proposed a system to identify the six emotion category proposed by Ekman et al. [14]. However, extracting more emotion category provides more insight about data. To the best of our knowledge, there is no machine learning approach of extracting nine emotion categories defined by Izard [6] from text data.

Although machine learning based approach of emotion extraction from text which could eliminate the problems of rule based approach, it was not taken into account. In that connection, we proposed a machine leaning based approach for the task. Our system relied on text labeled with emotion category rather emotion intensity lexicon and hence eliminate the tedious job of creating emotion intensity lexicon. Labeling a text with an emotional marker is comparatively easier than labeling a word with emotion intensity values. In our experiment, we used social media data from Twitter [11]. We collected 4000 tweets from twitter and labeled them with proper emotion category by human expert. We used some portion of them to train our system and the other to test. Among the various emotional states we used the nine basic emotion categories namely, ‘anger’, ‘disgust’, ‘fear’, ‘guilt’, ‘interest’, ‘joy’, ‘sad’, ‘shame’ and ‘surprise’ that can be identified by facial expression defined by Izard [6] along with ‘neutral’ category which stood for absence of emotional content.

The unique contributions of this work are: 1) We showed a machine learning based approach of extracting nine basic emotion categories from text which eliminated the necessity of emotion intensity lexicon; 2) We showed a comparative study on performance of three most common classifiers in the job of emotion extraction from tweets. This remainder of the paper is organized as follows. In the Section 2, we describe the methodology of this research. The outcomes of our experiment and the reasons behind them are discussed in Section 3. We conclude out work and also keep pointer to future directions in Section 4.

2. Research Method

In this work, we followed a supervised machine learning approach to identify the nine emotion categories proposed by Izard [6]. In supervised machine learning system, the system is trained with examples for which class labels are known. After the training, the system is tested on a data set with unknown class labels. The architecture of our system is shown in Figure 1. At first, we prepare our dataset to make it suitable for training and testing. Preparation of data is performed in the Data Preparation Module of our system. Emotion Extraction Module of our system is responsible for extracting emotion from processed data.

2.1. Data preparation module

In this module, data are processed so that they are ready to feed into the classifier algorithm. We collect data from social networking site Twitter because it is a great source of emotional text [11]. In this site, people express their emotion and opinion as a short form of text known as Tweet. Twitter API is used to collect data from Twitter. Collected tweets contained different types of noise. For this reason, we perform cleaning operation on the tweets. We eliminate re-tweets along with the tweets containing links to other multimedia content. After that, we manually label the tweets to one of the nine emotion categories defined by Izard [6]. Tweets containing no emotional content are added to 'neutral' category. Furthermore, all parts of data are not important for emotion extraction task. We apply preprocessing to eliminate uninteresting and trivial parts of the data. Preprocessing is very important step of classification because the accuracy of natural language processing system greatly depends on it [15].

Tokenization is the first step of preprocessing text. In tokenization, the text is chopped into a stream of words, phrases, symbols or other meaningful units called tokens. Along with decomposing text into tokens, tokenization may also eliminate some symbols such as punctuations. Stopwords are the token or term those occur frequently and are essentially meaningless in the sense that they contribute very less to the context and content of the text. Thus they are not important for classification of text and hence are eliminated. Normalization is a transformation process that transforms different representations of a word to one ideal form. For example, USA and U.S.A. are normalized to USA. Different variations of a word convey almost the same meaning and hence can be considered as the same word. We use stemming to convert variants of a word to the base form of that word.

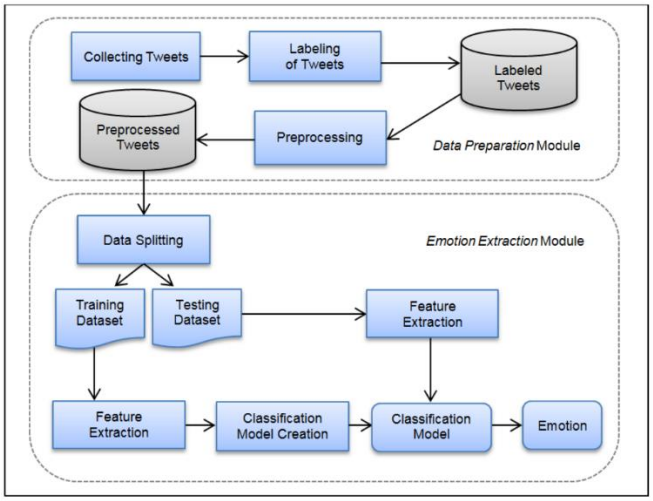

Figure 1. Architecture of our system

2.1. Emotion extraction module

The main task of emotion extraction from data is done by this module. In this module, to extract the emotional category of text, a classification model is created based on the training data. For this purpose, we extract important features from the text. We apply language modeling technique for feature extraction. We use Vector Space Model (VSM) and Probabilistic Language Model (PLM) for language modeling.

In VSM, a tweet is represented as a vector of terms in a common vector space. A dictionary is formed by collecting all the terms in the domain of discourse or corpus. Each term represents a direction and hence a unit vector in the document vector. For example, if a corpus consists of two tweets, T_1='*I am sick today*' and T_2='*I am angry*', then the dictionary formed by VSM modeling of the tweets is shown in Table 1.

Table 1. Dictionary formed by VSM modeling of the tweets

Tweet	I	am	sick	today	angry
T_1	1	1	1	1	0
T_2	1	1	1	0	1

The dictionary contains the count of each term in each of the tweets. The VSM representations of the tweets are as follows.

$$\overrightarrow{T_1} = 1\hat{I} + 1\widehat{am} + 1\widehat{sick} + 1\widehat{today} + 0\widehat{angry}$$
$$\overrightarrow{T_2} = 1\hat{I} + 1\widehat{am} + 1\widehat{sick} + 0\widehat{today} + 1\widehat{angry}$$

The number of occurrences or frequency of a term t in document d is called term frequency ($TF_{d,t}$). The $TF_{d,t}$ of the terms can be viewed as a quantitative representation of the document. This model is known as Bag of Words model in the literature of language modeling. However, $TF_{d,t}$ provides the importance of term to a particular document i.e., local weight. The problem of local weight is that it provides importance regarding a document rather the whole corpus. That's why we use global weight $TF - IDF_{d,t}$ which provides importance of a term according to the whole corpus. Global weight $TF - IDF_{d,t}$ is defined as,

$$TF - IDF_{d,t} = TF_{d,t} \times IDF_t$$

Where IDF_t is called Inverse Document Frequency. If there are total N number of documents in the corpus and DF_t represents the number of documents in the corpus that contain the term t then Inverse Document Frequency IDF_t is defined as,

$$IDF_t = log \frac{N}{DF_t}$$

We use probabilistic language modeling which provides the probability of a sequence of words. We use probabilistic N-gram modeling of language in our experiment. For a sequence of words, $w_1, w_2, w_3, \ldots\ldots w_i, \ldots\ldots w_n$ the Gram rule of language modeling is defined as,

$$P(w_1 w_2 w_3 \ldots . w_n) = \prod_i P(w_i | w_{i-k} \ldots . w_{i-1})$$

If the value of k = 1, then the language model is called bigram and for k = 2, the model is called trigram. Similarly, unigram model is defined as,

$$P(w_1 w_2 w_3 \ldots w_n) = P(w_1) \times P(w_2) \times P(w_3) \times \ldots\ldots \times P(w_n)$$

Unigram model doesn't consider the conditioning context that's why bigram modeling is used in our experiment. To deal with Zero Probability problem, we use Add-1 smoothing technique which adds 1 to each count so that no probability becomes zero. Important features extracted from the train data are fed into the classifier algorithm to create the classification model. We experiment Multinomial Naïve Bayesian (MNB) classifier, Support Vector Machines (SVM) and Artificial Neural Network (ANN) for classification model creation.

The Multinomial Naïve Bayesian classifier is a probabilistic classifier that predicts the class label based on the probability of a document to be of a particular class using Bayes' theorem. It assumes that the effect of an attribute on the class label is independent of the effect of other attributes [16]. The reason for using this classifier is that it found to faster that other classifiers [17]. Let, D denotes the training tweets and their associated class labels. Each tweet is represented by an n-dimensional attribute vector, $X = (x_1, x_2, x_3, \ldots, x_n)$. Here; attribute value x_i represents the measurement done on the attribute A_i. We have a total of n attributes $A_1, A_2, A_3, \ldots\ldots, A_n$ and m classes $C_1, C_2, C_3, \ldots., C_m$. The classifier will predict that a tweet X belongs to the class for which the posterior probability, conditioned on X is the highest i.e., the tweet X belongs to the class C_i iff,

$$P(C_i|X) > P(C_j|X) \; for \; all \; 1 \leq j \leq m, j \neq i.$$

Thus, to find the class label of X we need to maximize $P(C_i|X)$. According to Bayes' theorem,

$$P(C_i|X) = \frac{P(X|C_i)\,P(C_i)}{P(X)}$$

The probability $P(X)$ is constant for all classes, thus we only need to maximize $P(X|C_i)\,P(C_i)$. The $argmax$ function provides the class for which the probability is the maximum. Therefore, we need to find,

$$argmax\big(P(C_i|X)\big) = argmax\ (P(X|C_i)\,P(C_i))$$

The probability $P(C_i)$ can be calculated as,

$$P(C_i) = \frac{|C_{i,D}|}{|D|}$$

Where $|C_{i,D}|$ is the number of training tweets of class C_i in D. As the Multinomial Naïve Bayesian classifier assumes class conditional independence, the value of $P(X|C_i)$ can be calculated as,

$$\begin{aligned} P(X|C_i) &= \prod_{k=1}^{n} P(x_k|C_i) \\ &= P(x_1|C_i) \times P(x_2|C_i) \times P(x_3|C_i) \times \ldots\ldots \times P(x_n|C_i). \end{aligned}$$

We can calculate the probabilities $P(x_1|C_i) \times P(x_2|C_i) \times P(x_3|C_i) \times \ldots\ldots \times P(x_n|C_i)$ from the training tweets because, as stated earlier, x_k refers to the value of attribute A_k for tuple X.
To predict the class label of tweet X, $P(X|C_i)\,P(C_i)$ is calculated for each class C_i. The function $argmax$ will return the class C_i if and only if,

$$P(X|C_i)\,P(C_i) > P(X|C_i)\,P(C_i)\ for\ all\ 1 \le j \le m, j \ne i.$$

The value of the $argmax$ function is the emotional class of a tweet. The Artificial Neural Network is based on the modeling of neurons of human brain which can be considered as a weighted directed graph in which neurons represent the nodes and weighted directed edges can be considered as input and output connection of neuron [16]. A Neural Network is necessarily a connected collection of simple units called perceptron. Each link has a weight $w_{i,j}$ which defines the properties of the link. It has a bias input $a_0 = 1$ with weight $w_{0,j}$. Each perceptron j computes the weighted sum of the inputs and then applies a sigmoid function to generate the output. The Neural Network learns by iteratively processing each tweet of the training set. It calculates the error by comparing the output of the network with given target value. It than back propagates the error so that weight of connection links are adjusted to reduce the mean-squared error between the network output and actual target value.

The Support Vector Machine is a supervised classification technique in which maps the original data into a higher dimension space using nonlinear mapping. It then searches the higher dimension for a decision boundary which separates the tuple of one class from another which is called optimal separating hyperplane. SVM finds this hyperplane using support vectors. Support vectors are also training tuple that are closest to the decision boundary and pushed up by hyperplane [16]. The SVM is a binary classifier and its main function is to find a hyperplane, known as Maximum Marginal Hyperplane (MMH) which maximizes the margin between classes. A separating hyperplane can be written as,

$$W.X + b = 0$$

Where W weight vector of the form, $W = \{w_1, w_2, w_3 \ldots., w_n\}$; n is the number of attributes and b is a scalar quantity referred to as bias. The equation of the MMH can be written as a constrained convex optimization problem which can be solvable using Lagrangian formulation. After solving we find the following equation for MMH as decision boundary,

$$d(X^T) = \sum_{i=1}^{l} y_i \, \alpha_i \, X_i X^T + b_0$$

Where, y_i is the class label of the support vector X_i; X^T is a test tweet; α_i and b_0 are numeric parameters that are determined automatically by the SVM algorithm and l is the number of support vector. Given a test tweet X^T we put in it the equation above. If the sign of the result is positive then the tweet belongs to the positive class and if negative then to the negative class. To use SVM in our multiclass classification problem, we took one class and put all other classes altogether which is known as One vs. Rest. We applied the same procedure to all the emotion categories.

3. Results and Analysis

To evaluate our system, we split our dataset into disjoint train set and test set. The train set is used to train the system and the test is used to evaluate the performance of our system. We use the following three performance criteria to test the system.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$F\text{-}measure = \frac{True\ Positive + True\ Negative}{Positive + Negative}$$

We perform holdout method and k-fold cross validation for evaluation. For holdout method, we use one fifth of total data for testing and rest for training. We performed the test for three times and calculated the average of various measures which are depicted in Table 2.

Table 2. Evaluation results in holdout method of our system

Language Model	Classifier	Accuracy	Precision	Recall	F-Measure
Bag of Words	MNB	54.42	66	54	52
	ANN	41.25	46	41	41
	SVM	71.75	72	73	70
Bigram	MNB	55.85	67	56	53
	ANN	42.75	47	44	43
	SVM	74.20	73	74	71

It is clear from Table 2. That SVM performs better than the other two classifiers in all the performance criteria. However, the result of holdout method may be biased by a dataset which is not uniformly distributed. For this reason, we also perform 10-fold cross validation of our system. Total 4000 tweets of the corpus are divided into 10 folds of each containing 400 tweets. One fold is used for testing and other nine folds are used for training. The process is repeated ten times on all the folds. The result of the 10-Fold cross validation is shown Table 3.

Table 3. Evaluation results of the 10-Fold cross validation of our system

Language Model	Classifier	Accuracy	Precision	Recall	F-Measure
Bag of Words	MNB	52.98	66	52.9	46.6
	ANN	45.95	51.1	46.0	47.4
	SVM	75.23	75.7	75.2	71.7
Bigram	MNB	54.25	68	59	55
	ANN	47.25	53.50	47.25	48
	SVM	77.50	78	77.15	74.0

Comparing the result given in the Table 3, it is clear that in 10-fold cross validation also, Support Vector Machine outperforms the other two classifiers, Multinomial Naïve Bayesian and Artificial Neural Network algorithm. As there is not much difference in figures of holdout validation and 10-fold cross validation, it is intuitive that our dataset is uniformly distributed.

Moreover, SVM possesses high-dimensional input space and it is less affected by the overfitting problem. As like other text categorization problem, in our problem there are only a few irrelevant features and almost all features contain some considerable information that are necessary for emotion extraction. The number of features of our dataset in Bag of Words and Bigram modeling are 2754 and 8831 respectively. Because of this high-dimensional input space, SVM can capture all the related features.

Another aspect that is noticeable from the both Table 2 and Table 3 is bigram modeling provides better accuracy than Bag of Words modeling. This is because the Bag of Words modeling doesn't consider the context of a term in a tweet. However, the emotion of a term is also related to the previous terms of the tweet. In bigram modeling, the relation between two consecutive terms is also taken into account thus it provides better accuracy.

The reason of better accuracy of MNB classifier than that of ANN is that we used $TF - IDF_{d,t}$ weight and MNB performs better with it. On the other hand, the accuracy of ANN is lower than expected. We used default setup for ANN classifier. The accuracy may be increased by performing grid search method for optimal learning rate and changing the different parameters such as the number of hidden layers, number of iteration etc. of the network. Furthermore, increasing the size of dataset may increase the performance of all three classifiers.

4. Conclusion

Extracting emotion from text has a lot of important applications, such as e-commerce, product review analysis, personal or group level communication, automated motive detection for robotics etc. Automatically extracting emotion from text can be done by various approaches. A lot of rule based methods are established there. The main problem with these methods is creating a reliable emotion intensity lexicon with less possible effort. We analyzed supervised machine learning based approach to solve the problem. We experimented the Multinomial Naïve Bayesian classifier, Support Vector Machine and Artificial Neural Network in this research. In our setup, SVM outperformed other two classifiers with impressive accuracy for the task. Out work can be used to eliminate currently available tedious systems of creating emotion intensity lexicon. Although the text classification problem is very old, emotion extraction from text using machine learning techniques is comparatively new. There are many scopes of work in this sector. We experimented three supervised machine learning methods. Comparative study of others classifiers will be helpful for new researchers of this domain.

As a pointer to the future work, there is a lack of established dataset for 9 emotion categories proposed by Izard. Creating an established dataset will also be a great work in this domain. We used bigram model for semantic analysis. More complex semantic analysis model can be used with advanced feature extraction technique to increase the accuracy of the system.

References

[1] Asad MU, Afroz N, Dey L, Nath RP, Azim MA. *Introducing active learning on text to emotion analyzer*. In Computer and Information Technology (ICCIT), 2014 17th IEEE International Conference on 2014: 35-40.
[2] Abrams D, Hogg MA, editors. Social identity and social cognition. Oxford: Blackwell; 1999.
[3] Neviarouskaya A, Prendinger H, Ishizuka M. Textual affect sensing for sociable and expressive online communication. Affective Computing and Intelligent Interaction. 2007: 218-29.
[4] Picard RW, Picard R. Affective computing. Cambridge: MIT press; 1997.
[5] Strapparava C, Valitutti A. WordNet Affect: an Affective Extension of WordNet. InLREC 2004; 4: 1083-1086.
[6] Izard CE. Human emotions. *Springer Science & Business Media*; 2013.
[7] Dey L, Asad MU, Afroz N, Nath RP. *Emotion extraction from real time chat messenger*. In Informatics, Electronics & Vision (ICIEV), 2014 IEEE International Conference on 2014: 1-5.
[8] Afroz N, Asad MU, Dey L, Nath RP, Azim MA. *An intelligent framework for text-to-emotion analyzer. In Computer and Information Technology (ICCIT).* 2015 18th IEEE International Conference on 2015: 401-406.
[9] Subroto IM, Selamat A. Plagiarism Detection through Internet using Hybrid Artificial Neural Network and Support Vectors Machine. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2014; 12(1): 209-18.

[10] Pang B, Lee L, Vaithyanathan S. *Thumbs up? Sentiment classification using machine learning techniques*. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing. Association for Computational Linguistics. 2002; 10: 79-86.
[11] https://twitter.com/
[12] Susanti AR, Djatna T, Kusuma WA. Twitter's Sentiment Analysis on Gsm Services using Multinomial Naïve Bayes. *Telkomnika*. 2017; 15(3).
[13] Aman S, Szpakowicz S. Identifying expressions of emotion in text. In Text, speech and dialogue 2007: 196-205. *Springer Berlin/Heidelberg*.
[14] Ekman P. An argument for basic emotions. Cognition & emotion. 1992 May 1; 6(3-4): 169-200.
[15] Ruhwinaningsih L, Djatna T. A Sentiment Knowledge Discovery Model in Twitter's TV Content Using Stochastic Gradient Descent Algorithm. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2016 Sep 1; 14(3):1067-76.
[16] Han J, Pei J, Kamber M. Data mining: concepts and techniques. *Elsevier*; 2011.
[17] Sembodo JE, Setiawan EB, Baizal ZK. A Framework for Classifying Indonesian News Curator in Twitter. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2017; 15(1): 357-64.