Multiclass Emotion Extraction from Sentences

Bincy Thomas, Vinod P, Dhanya K A

Abstract— This paper aims to investigate the extraction of different classes of emotion from sentences using supervised machine learning technique, Multinomial Naïve Bayes (MNB). Here a bag of word approach is used to capture the emotions. The unigrams are mainly used for this and the bigrams and trigrams are used to capture lower order dependencies. The work is done on the ISEAR dataset [14]. The experiments with different feature sets selected using Weighted log-likelihood score (WLLS) [12] shows that the MNB classifier provides good results when the unigram feature set size is 450 which provides an average accuracy of 76.96% across all emotion classes.

Index Terms— Emotion identification, feature extraction, ISEAR dataset, MNB classifier, NLTK, WEKA tool, WLLS scheme.

1 Introduction

In human nature, emotions are an important and unavoidable element. The human emotions are widely studied in psychological and behavioural aspects. It is also an important domain in computer science. The emotions can be employed in affective computing. By analyzing the user's mood, computer systems can behave more intelligently in human-computer interactions.

The emotions can be specified through facial expressions, textual information, or speech. The widespread form of communication on web is in the form of text. The emotion inferred from this textual data is useful in many areas such as sentiment analysis, text to speech generation, and better computer interaction system.

Emotions may be expressed by a single word or a group of words. In computational linguistics, the automatic emotion detection from texts is becoming increasingly important from an applicative point of view. Emotion classification allows us to identify the feelings of individuals toward specific events. The most natural way for automatic emotion recognition of the user is to detect his emotional state from the text that he entered in a blog, an online chat site, or in another form of text.

The goal of this paper is to classify the emotions in text. It is focusing on identifying seven different classes of emotion such as, anger, disgust, fear, guilt, joy, sadness, and shame, using a lexicon-based approach. Unigrams are used as major features for the identification of emotions. To deal with the lower order dependencies, which unigrams fail to represent, bigrams and trigrams are also added. The text classification performs well with machine learning approaches. For emotion classification from text, the Multinomial Naïve Bayes approach is employed.

The remainder of this paper is organized as follows: Section 2 introduces related work in this area. The third section presents our proposed methodology. Section 4 is about the experiments and the results, which is followed by the conclusion. Last section is the references.

2 Related Works

There exist different categories of emotions. Ekman et al [10] classified the emotions into six basic categories: anger, disgust, fear, joy, sadness, and surprise, which are known as universal emotions.

In his proposed work, Jianhua Tao et al [9] generated emotion estimation net (ESiN) that combined the content words and emotion functional words (EFWs) to estimate the final emotion output. They used the text from a spontaneous speech corpus and obtained relatively good results.

Yang et al [7] proposed the emotion classification of web blog corpora using Support Vector Machine (SVM) and Conditional Random Field (CRF) machine learning techniques. They used the lexical terms appearing in a sentence as features for the classification and found that the CRF classifiers outperformed the SVM classifiers.

A bag of words approach to emotion classification was introduced by Danisman et al [6]. They considered the ISEAR dataset and tested various classifiers including SVMs, Naïve Bayes and Vector Space Model (VSM), and found that VSMs gave most promising results.

Knowledge and corpus based methods for automatic emotion identification was proposed by Strapparava et al [5]. They considered emotion analysis of news headlines, and proposed five systems for emotion extraction: WIN-AFFECT presence, Latent Semantic Analysis (LSA) single word, LSA emotion synset, LSA all emotion words, NB trained on blogs. They found that the LSA system using all the emotion words performed well.

Dipankar et al [4] proposed a mechanism for sentence level emotion identification. They used the Conditional Random Field (CRF) classifier for the classification of the words into the six emotion tags and one neutral tag. The presence of negative words had been handled by them effectively. They found an effective mechanism for sentence level emotion detection which can be used to identify document level emotion.

Bellegarda et al [2] described a method for emotion analysis based on the principles of latent semantics, using two techniques such as latent affective folding and embedding. They

- *Bincy Thomas is currently pursuing masters degree program in computer Science & Engineering with Specialization in Information Systems at SCMS School of Engineering and Technology, Karukutty, Kerala, India. bincythomas17@gmail.com*
- *Vinod P is working as an associate professors at SCMS School of Engineering and Technology, Karukutty, Kerala, India. pvinod21@gmail.com.*
- *Dhanya K A is working as an assistant professor at SCMS School of Engineering and Technology, Karukutty, Kerala, India.* dhannyashibu@gmail.com

ISSN 2229-5518

found that both latent affective folding and embedding techniques outperformed the standard LSA-based approach, and that the performance of latent affective embedding is slightly better than with latent affective folding.

A hierarchical approach to emotion recognition and classification in texts handling the unbalanced data was introduced by Ghazi et al [3]. They used a corpus of blog sentences that are annotated with emotion labels and found that the hierarchical approach produced better results for emotion analysis.

Chaffar et al [1] used a heterogeneous dataset collected from blogs, fairy tales and news headlines, for emotion analysis in texts. To find the best classification algorithm, they compared J48 for Decision Trees, Naïve Bayes for the Bayesian classifier, and SMO implementation of SVM, and used different features such as Bag-Of-Words (BOW), N-grams, and lexical emotion features. They found that SMO algorithm with BOW features performed better than others.

3 Proposed Methodology

This section describes the proposed method for emotion extraction from sentences. The features are extracted from the ISEAR dataset. The proposed system (refer Figure 1) selects features and perform classification using Multinomial Naïve Bayes classifier.

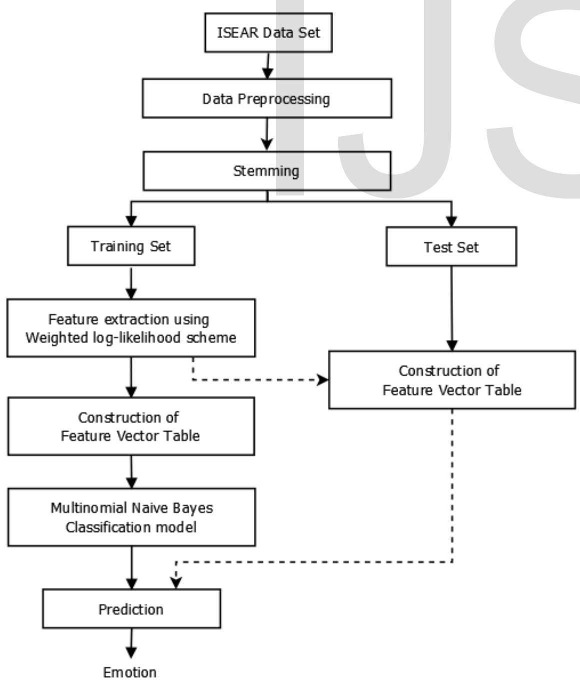

Fig 1. Proposed architecture for multiclass emotion classification

3.1 Dataset

Over a period of many years during the 1990s, a large group of psychologists all over the world collected data in the ISEAR project [14], directed by Klaus R. Scherer and Harald Wallbott. Student respondents, both psychologists and non-psychologists, were asked to report situations in which they had experienced all of 7 major emotions (anger, disgust, fear, guilt, joy, sadness, and shame). In each case, the questions covered the way they had appraised the situation and how they reacted. The final data set thus contained reports on seven emotions each by close to 3000 respondents in 37 countries on all 5 continents.

3.2 Data pre-processing

The ISEAR dataset is in the form of sentences which are tagged with the emotion experienced by the user, who are writing the sentence. There are seven emotions in the dataset: anger, disgust, fear, guilt, joy, sadness, and shame. The sentences in the dataset need to be pre-processed before performing any type of operations in it.

The dataset contains some useless statements such as "[No response]", "none", etc. Such statements are removed during the pre-processing step. Certain punctuations are important in identifying the emotions and others are of no use for emotion identification. The useless punctuations are removed during pre-processing. Exclamation mark (!) is replaced with special keywords XXEXLMARK and XXQUESMARK so that they can be used as features while training the classifier. In addition to that, the short forms of words which are represented with apostrophes are replaced to their true forms. For example, don't replaced to do not, I'm replaced to I am, etc.

Ngram features are used to train the emotion identification classifier. For that, the pre-processed data is first tokenized. But the ngrams obtained from tokenization is not so good for using as features for classification as they do not generalize well. This is mainly due to the presence of various forms (or inflections) of the same word (e.g., listener, listening, listened, etc are different forms of the word listen). To handle this situation, each word is replaced to their root form which is then used as features. Stemming is a technique for finding the root form of a word.

3.3 Stemming

Stemming is the process of replacing words (inflected words) to their root form or stems. For example, the words listener, listening, listened, etc are all reduced to their root word 'listen'. But the stemmed word may not be same as the morphological root of the word. For example, cookery is reduced to cookeri, which is not the actual word. In stemming, the ending characters of a word is just stripped to produce a stemmed form of the word rather than doing a dictionary look up to identify the actual root form of the word, which makes this technique computationally less expensive.

3.4 Feature extraction

Ngram features are found to be useful for text classification tasks. Here unigrams, bigrams and trigrams are used as features for emotion identification. The unigrams are found to be very useful features. These include adjectives, adverbs, verbs, and nouns. Bigrams and trigrams are used to deal with lower order dependencies, which the unigrams fail to capture (e.g., The show was barely interesting).

For extracting the features, the sentences in the dataset are tokenized to get unigrams, bigrams and trigrams. From these, the unigrams of length 2 or less are removed. Classification focuses on increasing the accuracy while reducing the feature length. Feature length is reduced by selecting some features from the available feature list using some selection criterion. Here we are using the weighted log-likelihood scheme proposed by Vincent et al [12]. In this scheme, every unigram, bigram, and trigram is assigned a weighted log-likelihood score (WLLS) with respect to each emotion. The WLLS for an ngram with respect to an emotion class is calculated as:

$$WLLS(w_i, c_j) = P(w_i \mid c_j) \log \frac{P(w_i \mid c_j)}{P(w_i \mid \neg c_j)} \quad (1)$$

where,

w_i represents the ngram (unigram, bigram, or trigram) whose score is to be evaluated.

c_j is the emotion class with respect whom the score is evaluated.

$P(w_i \mid c_j)$ is the ratio of count of the ngram w_i in the emotion class c_j to the count of all words in that class.

$P(w_i \mid \neg c_j)$ is the ratio of count of ngram w_i in the class $\neg c_j$ (all classes other than) to the count of all words in the class.

The *log* ratio in this scheme gives a high score to ngrams that are specific to each emotion class and a relatively low score to ngrams that are uniformly distributed across all emotion classes. So this WLLS scheme captures the relevancy of an ngram in each emotion class.

After scoring the ngrams with WLLS scheme, the ngrams are sorted in descending order of their scores. From this sorted list, top *u* unigrams, top *b* bigrams, and top *t* trigrams are selected as the features to train the classifier where *u*, *b*, and *t* are chosen empirically.

Table 1 shows the top 10 unigram and bigram features of the samples in the training set along with their corresponding Weighted log-likelihood Score (WLLS) belonging to the *joy* class.

TABLE 1
TOP 10 UNIGRAM AND BIGRAM FEATURES OF THE TRAINING SET AND THEIR WLLS SCORE

Unigram		Bigram	
Feature	**WLLS**	**Feature**	**WLLS**
Joy	0.573	when i	0.1927
happi	0.2935	select to	0.1761
select	0.264	wa select	0.1655
pass	0.2031	i pass	0.1635
glad	0.155	pass my	0.125
univers	0.1129	very happi	0.114
when	0.1118	had pass	0.1031
first	0.1111	that i	0.0936
accept	0.1036	a long	0.0901
long	0.1006	i reciev	0.0875

3.5 EMOTION CLASSIFICATION

For emotion identification from text, the multinomial implementation of *Naïve Bayes classifier* is used here. A Naive Bayes classifier is a simple probabilistic classifier based on applying Bayes theorem with strong (naive) independence assumptions. NB classifiers are used because it is fast, easy to implement and relatively effective [11] [12] [13]. *Multinomial Naive Bayes* (MNB) classifier is a specific instance of a Naive Bayes classifier which uses a multinomial distribution for each of the features.

Feature vector tables are constructed with the selected features for the seven emotion datasets: *anger, disgust, fear, guilt, joy, sadness,* and *shame*. These feature vector tables are used to build the classification model.

4 EXPERIMENTS AND RESULTS

4.1 Experimental setup

The ISEAR [14] dataset is used for conducting the experiment. The emotion tagged sentences in this dataset are first normalized by pre-processing. Then the sentences are tokenized for ngram extraction by using the tokenizer provided by Python NLTK [16]. Then the words are stemmed to their root form by stemming which is done by using the Porter Stemming [15] algorithm. The PorterStemmer module of NLTK is used for this purpose.

The whole dataset is then split into train set and test set in the ratio 50:50. The train set is then used to build the classification model and the test set is used to test the model. Testing is performed with samples not used for feature extraction.

Then WLLS scheme is applied to score the ngrams in the training set with respect to each emotion class. The ngrams are sorted in descending order of their scores which are used for training the Multinomial Naive Bayes (MNB) classifier. The MNB implementation of WEKA [17] is used for the classification. The classifier performance is measured for different feature sets.

4.2 Results

The test set is used to evaluate the performance of the classifier. The dataset is classified into seven emotion categories: anger, disgust, fear, guilt, joy, sadness, and shame using MNB classifier. The classifier accuracy is evaluated for different feature sets. The unigrams, bigrams, and trigrams are used as feature vector. The results are shown in table 2, 3 and 4.

TABLE 2
ACCURACY (%) OF VARIOUS UNIGRAM FEATURE SET

FL	400	450	500	600	700
A	68.33	68.94	68.83	69.25	69.92
D	69.72	69.76	69.67	70.3	70.45
F	82.65	**83.06**	73.63	73.78	73.88
G	80.35	**80.67**	70.34	71	71.47
J	77.41	**78.3**	70.25	70.27	70.79
Sd	78.94	**79.16**	73.1	73.2	73.66
Sh	78.8	**78.84**	66.53	67.2	67.77

TABLE 3
ACCURACY (%) OF VARIOUS BIGRAM FEATURE SET

ISSN 2229-5518

FL	200	400	500	600	700
A	37.71	40.47	43.24	44.07	45.89
D	**65.04**	29.39	32.74	33.75	33.84
F	**53.92**	37.09	36.55	38.71	40.99
G	**54.95**	35.66	37.78	37.84	38.7
J	**50.23**	37.84	38.83	40.23	40.58
Sd	**51.57**	32.41	35.29	37.84	38.52
Sh	**55.76**	43.34	45.75	44.83	45.68

TABLE 4
ACCURACY (%) OF VARIOUS TRIGRAM FEATURE SET

FL	100	200	250	350	450
A	15.23	16	17.79	18.76	20.45
D	18.58	19.01	19.93	20.35	20.51
F	20.41	22.02	22.99	25.63	25.56
G	18.83	20.1	20.79	22.22	24.17
J	22.44	23.29	23.75	25.16	25.66
Sd	19.25	20.78	21.53	24.07	24.88
Sh	18.88	23.91	24.28	24.97	26.86

FL : Feature Length
A : Anger Class
D : Disgust Class
F : Fear Class
G : Guilt Class
J : Joy Class
Sd : Sadness Class
Sh : Shame Class

4.3 INFERENCE

From the experiments, we can infer that for unigram features, except for two emotion classes (anger and disgust), the feature length of 450 gives better results than 700 feature length. In the case of bigram features, 200 feature set gives better results compared to 700 features set. This is because, as the feature size increases, the number of features with similar frequencies on all emotion classes (sparse features) also increases which cause the classifier unable to discriminate well between the classes.

5 CONCLUSION

This paper describes a bag of word approach for multi-class emotion extraction from sentences. The ngram features selected using weighted log-likelihood scheme, are used for testing the classifier accuracy. It provides good results when the unigram feature length is 450. The classifier accuracy decreases when the feature is increased which is due to the inclusion of more irrelevant features. Bigrams and trigrams provide comparatively less accuracy than unigrams. Thus we conclude that the unigram feature vector of length 450 is better for the multiclass emotion classification of sentences.

In future, we would like to evaluate the classifier performance on other publically available datasets such as SEMEVAL. Also we would further like to apply widespread range of feature selection techniques that could be used to create a robust emotion classifier.

REFERENCES

[1] Soumaya Chaffar and Diana Inkpen. “Using a heterogeneous dataset For Emotion Analysis in Text”, 2011.
[2] Jerome R.Bellegarda. “Emotion Analysis Using Latent Affective Folding & Embedding”, Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, 2010.
[3] Diman Ghazi, Diana Inkpen, and Stan Szpakowicz. “Hierarchical Approach to Emotion Recognition and Classification”, Canadian AI©Springer-Verlag Berlin Heidelberg, 2010.
[4] Dipankar Das and Sivaji Bandyopadhyay. “Sentence Level Emotion Tagging”, IEEE, 2009.
[5] Carlo Strapparava and Rada Mihalcea. “Learning to identify emotions in text”. In SAC. 2008.
[6] Taner Danisman and Adil Alpkocak. 2008. “Feeler: Emotion Classification of Text Using Vector Space Model”. AISB 2008 Convention, Scotland.
[7] Changhua Yang, Kevin Hsin-Yih Lin, and Hsin-Hsi Chen. “Emotion Classification Using Web Blog Corpora”, IEEE/WIC/ACM International Conference on Web Intelligence, 2007.
[8] Vincent Ng, Sajib Dasgupta and S. M. Niaz Ari_n. 2006. “Examining the Role of Linguistic Knowledge Sources in the Automatic Identification and Classification of Reviews”. Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions.
[9] Jianhua Tao. “Context Based Emotion Detection From Text Input”, National Laboratory of Pattern Recognition, 2004.
[10] Paul Ekman. “Facial Expression and Emotion”, American Psychologist, 48(4), 384–392. 1993.
[11] Harry Zhang. 2004. “The Optimality of Naive Bayes”. FLAIRS'04.
[12] George Forman Ira, George Forman and Ira Cohen. 2004. “Learning from little: Comparison of classifiers given little training”. 8th European Conference on Principles and Practice of Knowledge Discovery in Databases.
[13] Andrew Y. Ng and Michael I. Jordan. 2002. “On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes”. NIPS'02.
[14] "ISEAR Dataset" http://emotionresearch.net/toolbox/toolboxdatabase.
[15] "Porter Stemming" http://en.wikipedia.org/wiki/Stemming
[16] "Natural Language Toolkit (NLTK)" http://www.nltk.org/
[17] "WEKA" http://www.cs.waikato.ac.nz/ml/weka/