STM JOURNALS

Journal of Artificial Intelligence Research & Advances
ISSN: 2395-6720 (Online)
Volume 4, Issue 3
www.stmjournals.com

An Unsupervised Common Sense-based Learning Framework for Emotion Detection and Classification in Textual Social Data

Abid Hussain Wani**, Rana Hashmy
Department of Computer Science, University of Kashmir, Srinagar, Jammu and Kashmir, India

Abstract

Emotions are inherent to human behavior and generally expressed in response to the stimuli from the external environment or some internal processes. With the widespread use of online social platforms, people often express themselves using tweets, blogs and posts which generally include affective text. The topic of emotion detection and recognition has been widely studied in computational science and in fact a separate area "Affective Computing" is dedicated to its study. Although several techniques have been used for the purpose of detecting emotions from text, each technique has its own strengths and weaknesses. Our study focuses on employing lexical, semantic and real world information to detect six basic emotions from text. We propose a hybrid framework, for detecting and classifying emotions, which initially builds up the emotion lexicon, uses semantic information for grouping together related words and employs, a common sense platform, ConceptNet, to make the classification more accurate. The results obtained, utilizing this framework are quite encouraging and comparable to state-of-the-art techniques available.

Keywords: *Emotion detection, hybrid learning, affective computing, social data, real-world knowledge*

****Author for Correspondence*** E-mail: abid.wani@uok.edu.in

INTRODUCTION

Emotion detection and classification from text has received much attention especially from the people working on natural language processing and human-computer-interaction. In fact, since Picard put forward the idea of 'Affective Computing', researchers have been working on how to bring in a 'human feeling' into the machines so that they can recognize the emotional state on one side and behave and respond in turn with an appropriate 'emotional touch' [1]. With the huge proliferation of online textual resources (such as chatting messages, blogs, product reviews, emails, feedback posts etc.), identifying emotions from text has become ever more important particularly for human-computer-interaction. These resources comprising of user-generated content serve as a huge repository for the analyzing the textual content for the detection of users' emotional states [2]. Much of the research in affect detection has previously focused on opinion mining and sentiment analysis but in recent years, affect recognition at fine grained categorical level has received remarkable attention [3]. In this study, we attempt at detecting universally accepted six basic emotions i.e., anger, disgust, fear, happiness, sadness and surprise from text by employing word-emotion association lexicon known as EmoLex [4]. Next we employ word2vec to extract semantically similar word vectors. Our method uses a common sense platform, ConceptNet to make use of real-world knowledge to recognize emotions at sentence level [5].

RELATED WORK

The task of emotion detection and classification from text has been approached by researchers through a number of ways. Most of the initial work was based on simple keyword spotting techniques, which was later enriched with use of language resources to get away with certain inherent weaknesses in this approach [6]. Lexical resources such as WordNet, WorNetAffect, and SentiWordNet have been used in various studies for affect detection [7, 8]. Many studies have also been carried out which in addition to lexical resources use supervised learning algorithms such as support vector machine for emotion-directed classification text [9, 10]. Although the results achieved in utilizing language resources with supervised learning are quite satisfactory albeit the fact that supervised learning calls for annotation of a large corpus by humans, which is a tedious process and generally does not scale well from one domain to another. Our study attempts to get rid of this manual annotation by employing emotion-annotated lexical resource and semantic features of the language for unsupervised learning of emotion categories from the textual corpus. The results of the proposed methodology are satisfactory and scale well to varied domains while doing away with the need of manual annotation, although the supervised technique may give better results in certain cases. Although the utility of real-world knowledge for affect detection has been studied in some works, yet our study is the first of its kind to use semantic information alongside real-world knowledge to recognize the basic set of emotions from text [11, 12].

LEXICAL, SEMANTIC AND KNOWLEDGE RESOURCES

Our system for detecting and classifying emotions embarks on the use of two important resources which drive our unsupervised learning algorithm, EmoLex and ConceptNet. A brief introduction of each is presented below:

Emolex

In our study, we employ the emotion associated lexical resource EmoLex (also known as NRC Word-Emotion Association Lexicon) [4]. For EmoLex, the annotations have been manually done by crowdsourcing using Mechanical Turk. EmoLex is a list of English words and their associations with eight emotion categories (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive). In our study, we take into account only those words which have the associated emotion from any of six basic emotion classes i.e., anger, fear, surprise, sadness, joy (happiness in study), and disgust. We consider only six basic emotions in our study as Ekman argues that all other emotions are basically a blend of two or more of these basic ones [13]. This lexicon comprises of 14,182 unigrams (words).

ConceptNet

We used the semantic network from MIT Media Lab named as ConceptNet, which uses the database from the Open Mind Common Sense project to extract properties common to word clusters [14]. ConceptNet is a knowledge representation project, encompassing a large semantic graph that attempts to describe general human knowledge. This description of knowledge is done in natural language. ConceptNet includes description of words and common phrases and how they are related in common sense. It offers a large set of background knowledge that a computer application can employ as a guide for how the words and phrases are related to each other not only by their lexical definitions but also through common knowledge. ConceptNet currently supports 10 core languages with all API features and 68 more common languages, with vocabularies of at least 10,000 terms. Semantic relations between words are expressed by means of predicates or properties. Language terms or words are related between each other with relations such as MotivatedByGoal, CreatedBy, CapableOf, HasProperty, Causes etc. Relations can be prefixed with "Not" to express a negative assertion such as NotPartof, NotHasProperty.

Word2Vec

Word2vec is a computationally-efficient two-layer neural net based predictive model for learning word embeddings from raw text. Word2vec takes raw text corpus as input and its output is a set of vectors which are feature vectors for words in that corpus. Its applications extend well beyond parsing sentences in the wild and can be applied just as well to social media graphs, codes, genes likes, playlists, and any other series in which patterns may be found. Word2vec essentially treats words as discrete states like and looks for the transitional probabilities between those states i.e., the likelihood that they will co-occur. The principle aim of Word2vec is to group the vectors of similar words together in vector space, which means that it detects word similarities mathematically. Word2vec generates vectors that are essentially distributed numerical representations of word features, features such as the context of individual words. The beauty of Word2vec lies in the fact that it does so without any human intervention. Therefore given enough data, usage and contexts, Word2vec can make highly accurate guesses about a word's semantics based on its past appearances, which can be used to establish a word's semantic-relatedness and association with other words (e.g. "principal" is to "school" what "ceo" is to "company"), or cluster documents and classify them by topic. The output of the Word2vec neural net is a dictionary wherein each item has a vector attached to it. The vector for each word can be fed into a deep-learning net or simply queried to discover associations between words. In our study, we utilize this dictionary to cluster words as per their affect sense.

METHODOLOGY

Although a number of approaches for detecting fine-grained emotions from text are available, yet most of them fail to robustly recognize affect and make the classification regardless of the domain from which the sentences are processed. Our argument is that the ability to sense differences in affect can be augmented by using common-sense knowledge. We suggest utilizing large-scale real-world knowledge to undertake the task of textual affect-sensing that not only provides a fine-grained emotion classification but also addresses many of the robustness and size-of-input issues associated with contemporary approaches. Our approach provides high degree of robustness by not only taking into account the surface features of the text like in keyword spotting, but also evaluates the affective qualities of the underlying semantic content of text using both semantic and real-world information. The fact that real-world knowledge is utilized for detecting affect equips our framework to sense the emotions from the text even when affect-bearing words are absent.

Given the fact that statistical NLP techniques which are semantically weaker often require a good number of sentences as input for reasonable accuracy; we propose a semantically rich and commonsense knowledge based approach which can sense emotions on the sentence-level without requiring a big chunk of sentences for reasonable accuracy. Though some studies have used hand-crafted models for the purpose but they usually tend to have limited coverage, require deeper semantic understanding and do not scale well to different domains. In our study, we employ a large-scale commonsense knowledge platform so as to avoid requirement for hand-crafting knowledge and take advantage of the greater coverage of the knowledge resource which ultimately makes our approach more robust.

Since our approach models affect at the sentence level and takes into account semantic and conceptual level, therefore there are little chances of our classification framework being tricked by structural features like ambiguity and negation at the word-level. Moreover, the fact that ConceptNet, our commonsense knowledge resource is basically an embodiment of a large body of real-world knowledge covering diverse domains makes it suitable for varied domains. The approach followed in this paper embarks on the notion that there is some user and domain-independent similarity in people's affective knowledge of emotions and attitudes towards everyday happenings which is somehow connected to people's view and commonsense about the world.

Having dwelled into the guiding idea of our approach in terms of existing approaches and theoretical considerations, the rest of this section will focus on more practical considerations, such as how lexical, semantic and commonsense resources are utilized and integrated for detecting emotions from textual data. Figure 1 depicts the proposed framework.

Overview of the Framework

Emotion detection from text is modeled as a classification problem and a sentence can be tagged with any of the seven classes (six classes for basic emotions and one for neutral). We build our classification system using lexical resources and common sense knowledge and the subject, the target sentences to emotion detection.

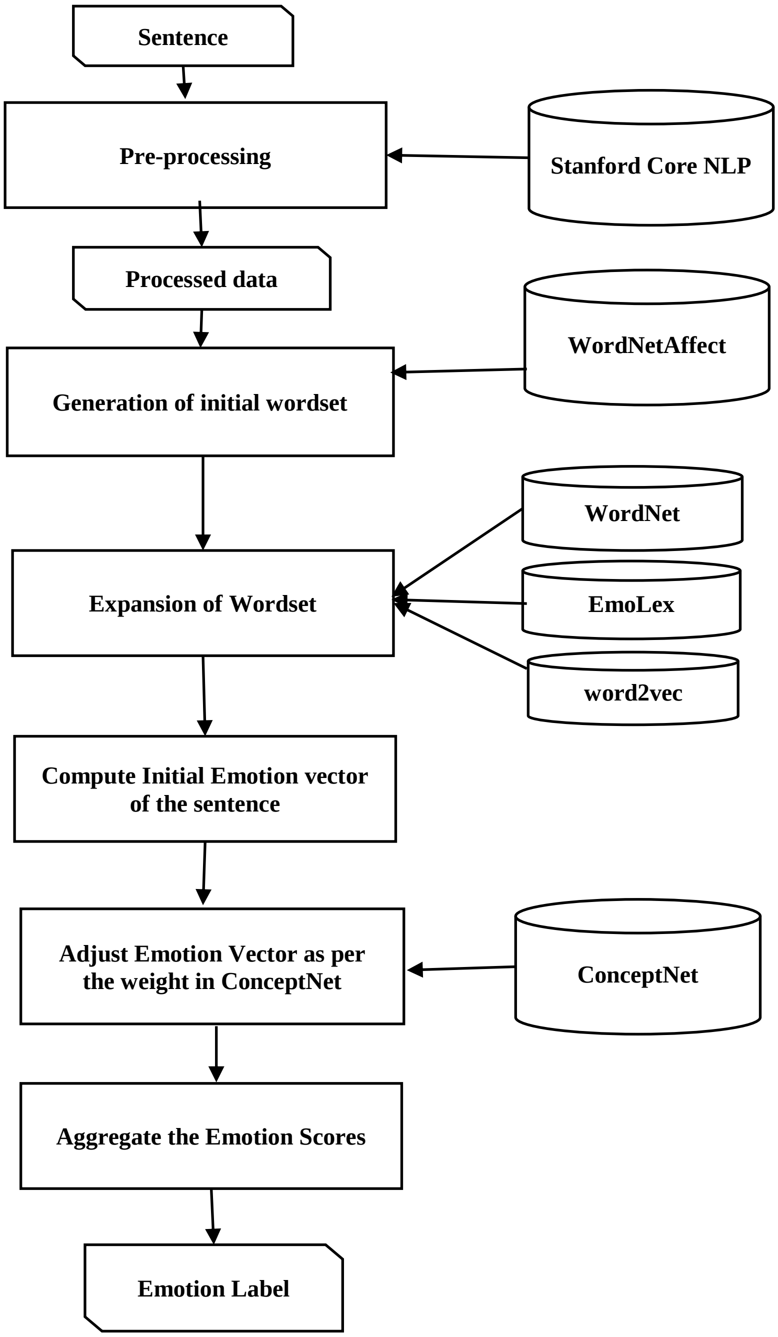

Fig. 1: *Proposed Framework.*

The target sentences are preprocessed before being looked in for affect sense. In the preprocessing step we use StanfordCoreNLP [15]. Preprocessing encompasses tokenization, POS tagging, lemmatization, stopword removal and slang mapping. Once the sentence is preprocessed, an emotion vector is used for evaluating the emotional affinity of words within the sentence using lexical resources, which provide initial input to the emotion vectors. Adjustments to these emotion vectors are then performed using real-world knowledge before arriving at the emotion label for a sentence.

Grouping of Lexically and Semantically Similar Words into Wordsets

Words which are lexically and semantically connected to each other are organized into separate files for each emotion category. This file contains two fields: the word itself and its weight. Theses weights are used to populate the emotion vector. We use WordNetAffect and EmoLex to capture the lexical relatedness and word2vec to capture the semantic affinity. The emotion vector is first populated by the weights assigned to the words in each wordset. The weights assigned in this study are as under:

WordNetAffect weight (w_{wn}): Words extracted for each emotion category from WordNetAffect are assigned a weight of 1.

WordNet weight(w_w): WordNet is used to get synonyms of the representative words for each emotion extracted from WordNetAffect. These synonyms are assigned a weight of 0.9.

Emolex weight(w_e): Words extracted from NRC emotion lexicon receive a weight of 0.7 for each category.

word2vec weight(w_{wv}): We train the word2vec model using Wikipedia text dump. Words in the vector for each catgory are assigned a weight of 0.6.

In order to deal with duplication of word in different wordset-weight files, we keep only the word which has highest weight and remove all of its other duplications. All the above weights have been assigned empirically.

Utilization of Common Sense Knowledge

Once we build our lexical resources, we adjust the emotion vectors leveraging the real-world knowledge from ConceptNet. Although ConceptNet for emotion detection has been used in a study, but this is the first study to use the assertion weights of ConceptNet for emotion detection [16]. Assertion weight expresses the strength of an assertion. We represent the ConceptNet weight of a phrase by wc. Table 1 gives an illustration of the assertions and their weights.

Table 1: *Assertions with Weights.*

Term	Relation Name	Related To	Relation Weight
taking finals	Causes	success	2.83
following a recipe	Causes	success	1.0
jumping at chance opening	Causes	success	1.0
a business	Causes	success	1.0
praying	Causes	success	1.0
punch someone	MotivatedByGoal	hurt	3.46
check vital signs	MotivatedByGoal	hurt	2.0
call for help	MotivatedByGoal	hurt	1.0
cry	MotivatedByGoal	hurt	1.0
stab to death	MotivatedByGoal	hurt	1.0
ignorance	CreatedBy	fear	1.0
mental illness	CreatedBy	fear	1.0
perceived danger	CreatedBy	fear	1.0
animals in the jungle	HasProperty	scary	1.0
Some movie plots	HasProperty	scary	1.0
red skys	HasProperty	scary	1.0
	HasProperty	scary	1.0
an event	CapableOf	surprise	2.0
a loud noise	CapableOf	surprise	1.0
some questions	CapableOf	surprise	1.0

Table 2: *Phrase-Emotion Relation.*

Term	Relation Weight	Suggested Emotion
taking finals	2.83	happy
following a recipe	1.0	happy
jumping at chance opening	1.0	happy
a business	1.0	happy
praying	1.0	happy
punch someone	3.46	anger
check vital signs	2.0	anger
call for help	1.0	anger
cry	1.0	anger
stab to death	1.0	anger
ignorance	1.0	fear
mental illness	1.0	fear
perceived danger	1.0	fear
animals in the jungle	1.0	fear
Some movie plots	1.0	fear
red skys	1.0	fear
an event	2.0	surprise
a loud noise	1.0	surprise
some questions	1.0	surprise

We retrieve the affect-bearing phrases for each emotion category from ConceptNet using different ConceptNet relations and store them in six different files (one for each emotion category). It is noteworthy to mention that this process effectively brings-in the real-world knowledge into our system, thereby making the classification more powerful and precise. Table 2 depicts this phrase-emotion relationship.

Computation of Emotion Vector

The emotion vector is computed at two levels: word and sentence level.

Emotion Vector for Words in a Sentence: The emotion vector for each word contained in a sentence is computed by summing the weight of word from three lexical files, i.e., from WordNetAffect, WordNet and EmoLex (each with associated weight). Suppose w_l is the weight from this weighted lexicon then we have,

$$w_l=w_{wn}+w_w+w_e+w_{wv} \tag{1}$$

To calculate the emotion vector of whole sentence, we take the summation of all affect words, as shown in Eq. (2).

$$S_l=w_{wn}+w_w+w_e+w_{wv} \tag{2}$$

In order to further improve our classification framework and make it more robust, we use the common sense knowledge from ConceptNet. The weight of the sentence is adjusted to capture the real-world knowledge before taking a final call on emotion category. From ConceptNet knowledge base we extract all relevant phrases for each emotion category specified by a relation. If m is the number of phrases which find a match from affective assertions retrieved from ConceptNet, then w_c, assertion weight, is calculated by aggregating assertion weights of all matched phrases and taking their average as shown in Eq. (3):

$$w_c=(\sum_{i=1}^{m} Pi\ )/n \tag{3}$$

Where, Pi is the weight of ith phrase matching the assertion.

The final value of emotion vector for a sentence is computed as shown in Eq. (4):

$$S=S_{l+}w_c \tag{4}$$

After obtaining the emotion vector $S=<s_1, s_2, s_3, s_4, s_5, s_6>$ of a sentence, if the highest score is above a certain threshold t, the sentence is labeled with that emotion. Otherwise, it is classified as neutral. The final emotion label L_S is computed as:

$$L_S=\begin{cases} E_k \ if\ S > t\ and\ max\ (s_i) = s_k \\ Neutral\ otherwise \end{cases} \tag{5}$$

EVALUATION AND RESULTS

As is the standard practice with classification problems to empirically analyze their classification performance, there are three threshold values. We calculate these values to measure performance of experimental results.

Precision rate (P) = Number of correctly-classified items/Number of all items put in this class

Recall rate (R) = Number of correctly-classified items/Number of all correct items in this class

F1 is the balanced value of P and R to evaluate the overall result of classification.

F1=2×P×R/(P+R)

To evaluate our framework for emotion detection we employed two standard datasets ISEAR and Aman Corpus [9]. The ISEAR (International Survey on Emotion Antecedents and Reactions) dataset is publicly available and contains a large number of personal reports on situations related to seven emotions i.e., joy, fear, anger, sadness, disgust, shame and guilt, collected from more than 3000 students from all over the world. We have used in our study, a subset of the ISEAR data that pertains to emotion categories of our interest: happy, fear, anger, sadness, disgust and surprise. We also used Aman's Annotated Emotion corpus to evaluate our framework. The results obtained appear in Table 3.

Table 3: *Evaluation Results of Proposed Framework.*

Emotion Category	ISEAR Dataset			Aman's Dataset		
	P	R	F1	P	R	F1
anger	47.63	52.47	49.93	56.23	62.5	59.20
disgust	53.23	58.12	55.57	48.30	51.23	49.72
fear	59.22	55.47	57.28	52.46	67.33	58.97
happy	68.66	76.98	52.58	71.23	65.85	68.43
sad	56.30	62.45	59.22	48.56	55.75	51.91
surprise	46.25	51.22	48.61	45.67	62.33	52.72

The results obtained indicate that our framework on the whole yields better results for Aman's dataset than ISEAR dataset. For

ISEAR dataset, the emotion sad is detected with highest accuracy, fear, disgust and happy emotions are also detected with a good accuracy while anger and surprise emotions are detected with least accuracy. In case of Aman's dataset, the performance is different for most of the emotion categories as compared to ISEAR dataset. The classification accuracy is best for emotion happy than any other emotion category. In sharp contrast to ISEAR dataset, our classification approach yields better results for anger, fear and surprise emotions.

CONCLUSIONS

We introduced an emotion detection and classification framework which exploits the lexical, semantic and real-world knowledge to detect affect in text. Though the utility of lexical and semantic resources for affect sense detection has been demonstrated in many earlier studies, yet our study stands as unique in the area of affect detection as it combines lexical and semantic approach along with utilizing real-world knowledge. Often, people express emotions through descriptions of situations without using affect bearing words and thus affect can be recognized only by utilizing common-sense knowledge which will guide us to the specific emotions triggered. Automatic detection of emotions is an even more challenging task to tackle, as it requires methods to gather, integrate and generalize the commonsense knowledge, on the basis of which, specific emotions can be detected and classified from textual data. In case of ConceptNet, even though a relatively small number of properties have been exploited by us, yet the amount of knowledge that we have gathered has equipped us to infer a large number of new concepts that trigger emotions. Our results confirm that the affective link between situations or concepts and the specific emotions triggered is given by the common relations, which in turn map into a specific emotion category. The results of evaluations show that the framework we proposed yields comparatively more accurate results than other recent approaches utilizing real-world knowledge. In future, we would like to extend the set of emotions to be detected to have a more fine-grained emotion classification.

REFERENCES

1. Picard RW. Affective Computing: Challenges. *Int J Hum Comput Stud*. 2003; 59(1–2): 55–64p.
2. Liew JSY, Turtle HR, Liddy ED. EmoTweet-28: A Fine-Grained Emotion Corpus for Sentiment Analysis. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*. 2016.
3. Clavel C, Callejas Z. Sentiment Analysis: From Opinion Mining to Human-Agent Interaction. *IEEE Trans Affect Comput*. Jan 2016; 7(1): 74–93p.
4. Mohammad SM, Turney PD. Crowdsourcing a Word-Emotion Association Lexicon. In *Comput Intell*. 2013; 29(3): 436–465p.
5. Speer R, Havasi C. Representing General Relational Knowledge in ConceptNet 5. *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*. 2012; 3679–3686p.
6. Thelwall M, Buckley K, Paltoglou G, *et al*. Sentiment Strength Detection in Short Informal Text. *J Am Soc Inf Sci Technol*. 2010; 61(12): 2544–2558p.
7. Valitutti A, Strapparava C, Stock O. Developing Affective Lexical Resources. *PsychNology Journal*. 2004; 2(1): 61–83p.
8. Esuli A, Sebastiani F. SentiWordNet: A High-Coverage Lexical Resource for Opinion Mining*. *LRE.tex*. 2015; 19(9).
9. Aman S, Szpakowicz S. Identifying Expressions of Emotion in Text. In *Text, Speech and Dialogue (TSD)*. 2007; 4629: 196–205p.
10. Strapparava C, Mihalcea R. Learning to Identify Emotions in Text. In *Proceedings of the 2008 ACM Symposium on Applied Computing (SAC '08*). 2008; 1556p.
11. Liu H, Lieberman H, Selker T. A Model of Textual Affect Sensing Using Real-World Knowledge. In *Proceedings of the 8th International Conference on Intelligent User Interfaces (IUI'03*). 2003; 125p.
12. Balahur A, Hermida JM, Montoyo A. Detecting Implicit Expressions of Emotion in Text: A Comparative Analysis. *Decis Support Syst*. 2012; 53(4): 742–753p.

13. Ekman P. An Argument for Basic Emotions. *Cogn Emot*. 1992.
14. Liu H, Singh P. ConceptNet: A Practical Commonsense Reasoning Tool-Kit. *BT Technol J*. 2004; 22(4): 211–226p.
15. Manning C, Surdeanu M, Bauer J, *et al*. The Stanford CoreNLP Natural Language Processing Toolkit. *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. 2014; 55–60p.
16. Balahur A, Tanev H. Detecting Implicit Expressions of Affect from Text using Semantic Knowledge on Common Concept Properties. In *10th International Conference on Language Resources and Evaluation, LREC 2016*. 2016; 1165–1170p.

Cite this Article

Abid Hussain Wani, Rana Hashmy. An Unsupervised Common Sense-based Learning Framework for Emotion Detection and Classification in Textual Social Data. *Journal of Artificial Intelligence Research & Advances*. 2017; 4(3): 49–56p.