Identifying and sensing emotional quotient weightage in the outcome using Speech Dialogue

Rajat Yadav, Anurag Mishra
ABES Engineering College, Ghaziabad

Anurag.mishra@abes.ac.in

Abstract. Emotion is a very important part of any interaction whether it is human-human or human-computer interaction as an emotion conveys the essence of the interaction. Decision making is also influenced by the emotional state of a person. If we can capture emotions from a human-computer interaction then it will be very beneficial for the computer to learn from that interaction. It can be efficiently used by an artificial intelligence system to understand and give response accordingly. The method proposed in this paper uses a machine learning based approach to detect emotion from the text obtained from the speech dialogue.

1. Introduction

An emotion describes the feeling of a person which can be expressed in more than one way like when a person communicates then the tone of the speech signifies a lot about the emotional state of the person at that time. In speech synthesis a lot of factors have to be taken into consideration to detect the emotion. Detecting emotion using speech synthesis is a different domain and since our focus is to detect emotion from text which will be generated from the speech, we will limit ourselves to the emotional analysis of text. Six primary emotions were proposed by a psychologist named Paul Ekman which are anger, disgust, fear, happiness, sadness and surprise. Later a professor named Robert Plutchik proposed eight primary emotions that are anger, fear, sadness, disgust, surprise, anticipation, trust, and joy. Plutchik also created a wheel of emotions which demonstrates that emotions can be expressed at varying intensities and different emotions can be formed when they mix with one another. The objective of this paper is to detect emotion from a text on the basis of Plutchik’s wheel of emotions using a machine learning approach. This paper uses a web based GUI (Graphical User Interface) to record speech of the user and then saves it in the database for further processing. To convert the recorded speech into text IBM Watson Speech to Text API is used. To use any machine learning model, it has to be trained first so for training the model the dataset from CrowdFlower is used which consists of 2550 sentences. In the dataset each sentence has been assigned an emotion out of seventeen emotion responses used for annotation by contributors.

The sentences and associated emotions were extracted from the dataset for preprocessing which uses concepts of NLP (Natural Language Processing), which is a field that describes the interactions between computer and human languages. The preprocessing of data is important as textual data has to be converted to mathematical data so that it can be used by any machine learning model. After training
the model it is used to predict a single emotion label for the text generated from the speech. The detected emotion is then displayed as the output to the user.

2. Related Work

Emotion classification on the basis of text is usually based on supervised machine learning techniques which uses training data that is labeled. Two types of labeled datasets are: manually labeled and pseudo labeled. The quality of manually labeled dataset is high since each sample is annotated by multiple humans. Pseudo labeled dataset is generated by tagging social media posts with emoticons. To retain the high quality with smaller sized datasets an ensemble method is used that uses both a bag-of-words (BOW) based linear model and a pre-trained word vectors based non-linear model. This is the first research paper where use of pre-trained word vectors was analyzed for improvement in emotion detection from text. A technique CLASS was introduced which can represent a document as a dense vector based on the significance of words in the document. By using the CLASS technique with combination of BOW and embedded representations in an ensemble, better results were obtained in datasets belonging to particular domain in comparison to older methods. The discussed method requires very less computing power and fits a smaller number of model parameters as compared to various deep-learning approaches. [1]

Keyword spotting technique has been applied extensively in the domain of speech and emotion recognition. This study has combined both keyword spotting and semantic analysis for emotion detection in blog reviews. A tool known as RapidMiner which is a powerful analytics platform was used to obtain the texts from review websites. Due to the limitations of keyword spotting approach, semantic analysis was used to match the words with the appropriate emotion with the help of WordNet dictionary. Synonym sets were used to link the relations with the help of classifiers. Classification algorithms like K-Nearest Neighbor (KNN), Naïve Bayes and Support Vector Machine (SVM) were used in the analysis. The study involved half a year of reviews of hotels and restaurants in blogs in the Philippines. For validation of results annotation was done by experts in the post-processing stage. The developed system produced a low result because of the small training dataset containing only over 1000 combinations. As the blog reviews were written by professionals, they were technical in nature and they contained words which did not produce any emotion. Due to this, neutral emotions were collected mostly. Results were better with 1000+ combinations as compared to 200 combinations. It would be very useful to include dictionaries of proper nouns to identify words and to use adjectives and adverb to modify the semantics. Using blog with more emotional content such as blogs on human relationships was recommended. [2]

Various approaches are being used for emotion recognition from text such as learning based method, lexical affinity and keyword spotting. This paper proposes an architecture based on spotting keywords and ontology of emotions. The use of ontology makes this model more effective in recognizing textual emotions. An ontology describes the concepts and relationships that may exist for an agent or an agent community. Ontology have definitional aspects like high level schemas and aspects like entities and attributes interrelationship is between entities, domain vocabulary. The emotion detector algorithm given in this paper compares the calculated scores for every emotion class of the primary level with their respective secondary and tertiary level classes and emotion class with the maximum score is assigned as the emotion of the input text. Each text which is used as an input is manually assigned an emotion class and then compared with the emotion class produced as output by the proposed emotion detection system. The proposed system was able to predict correct emotion class for 116 blogs out of 135 blogs. This paper explores the prevailing research of emotion recognition in textual data and proposes an emotion recognition system for improving emotion detection abilities in an efficient way with the help of ontology. [3]
Many emotion detection methods use supervised learning techniques which require a lot of annotated data for training the model. The methods which use dictionaries of emotion keywords and use an affect lexicon-based approach are limited to specific number of categories of emotions in the dictionary.

This paper suggests an unsupervised approach that recognizes the emotion based on context without the help of affect dictionaries and training data which is annotated. In the beginning a small set of representative terms are decided that are used for determining an affect-bearing word's emotional vector by measuring the semantic relation between the word and the emotion. Three types of syntactic dependencies are used for deciding the context of the word to enhance the emotion vectors. In this paper an unsupervised technique which is context sensitive is used to identify emotions from text. This method does not require any data set which is annotated or any affect lexicon. Results indicate that the method was more accurate than other latest unsupervised techniques and results were similar to some of the supervised techniques. One of the shortcomings of this method is that the Semantic Relationship scores depend on the source of the text corpus from which they are obtained. The Wikipedia corpus was observed to be better than the other two corpora and the context-based technique outperformed the context-free technique supporting the usefulness of words in the context.[4]

The approach used in this paper can assess the level of anger, fear, joy and sadness in a text. The aim here is to understand the meaning of the emotions observed and to identify the inferences of the obtained results. It is shown that the proposed approach is able to identify distinct emotional patterns in texts of the same literary genre like stories for children, the general emotional propensity of the writer, variations in the texts of the same writer, variations in the emotions expressed by semi-formal writings. In this paper the emotional analysis done on some stories of children from five different writers has been presented. Around 20 writings of two popular Spanish writers from different eras: Gustavo Adolfo Bécquer (1836-1870) and Mario Benedetti (1920-2009) were analyzed in each category of prose and poetry to recognize a generic emotional profile for both writers. It can be inferred that the identification of emotions in texts is an efficient way of increasing user models in various contexts such as e-learning and help desk systems. Another function possible for the method discussed is to construct a given user's emotional profile by analyzing the emotional pattern of the user's texts. After the profile has been developed the system could track and raise an alarm when a text has an emotional signature significantly different from the regular pattern of the user.[5]

The first goal of this paper is to detect different predefined speech styles on the basis of only the text content of human-computer interaction. The second goal is to use the recognized speech style in the dialogues between the intelligent entities for signing aspects such as speech style or other nonlinguistic aspects. The third goal is to recognize the emotion of the participant in the communication. The system described in this paper is implemented in the Hungarian language. For training the text a small set was gathered comprising of emotions such as sadness, happiness, anger and surprise. Little corpus was also gathered for the speech style. The tests show that both the speech styles and the emotions have been recognized well. The solution for the speech style classification has been implemented in the Java programming language. A test bed known as VirCA (Virtual Collaboration Arena) which is a modular, virtual test bed with 3D functionality was used for our system. VirCA’s network communication is based on OpenRTM software platform. The system used in this paper is integrated into this platform as RT-Component (Automatic Text Classification component). The application includes spoken dialogue systems, audio-visual games, vehicle-human interaction.[6]

As compared to emotion recognition in prosody, physiological state and facial expressions there is a lack of work on emotion recognition from text. The obligation to understand emotions is supported by a report "W3C is to work on a web standard for Emotion Markup". The approaches such as keyword based, learning based and hybrid based presently dominate the emotion detection task. Mainly syntactic features such as phrase patterns, pos tags, n-grams and semantic features like synonym sets are used by
these approaches for emotion detection. This paper exhibits that in what ways models are built on the basis of emotion theories and how those models are implemented in computational techniques for detecting emotions. This paper demonstrated a literature survey of the recent research in this field. A hybrid approach for emotion detection is proposed in this paper that shows that syntactic and semantic knowledge can greatly improve the prediction accuracy which is justified with experimental results. In training 2340 sentences were used and in testing out of 560 sentences LIBSVM correctly classified 540 sentences.[7]

This paper focuses on recognizing emotions from short blog texts in which the author’s mood is not described explicitly. This paper explores which emotional signals are accessible in a computer-mediated environment and measures the effects of top-down content analysis through specific emotion and linguistic categories. Additionally, data-driven techniques are applied to classify opinion and mood. Emotional linguistic features were explored in blog-text samples of 50 and 200 words. It was discovered by using automated content analysis (LIWC) that angry authors use a lot of negative effect words and happy authors use more positive effect words. The LSA method which is a semantic space co-occurrence technique was used to classify fear texts. The combination of the discussed methods may enable more accurate recognition of finer grained emotions.[8]

The field of emotion recognition and affective analysis can provide valuable contribution in different scenarios like computer assisted creativity, sentiment analysis, verbal expressivity in human computer interaction. The purpose of this paper was to recognize emotions from news headlines. Several algorithms have been discussed in this paper ranging from a simple heuristic algorithm to advanced algorithm. The simple heuristic algorithm directly checks for specific affective lexicons and the advanced algorithm checks for sameness between emotions’ representations in the latent semantic space and use Naïve Bayes classifiers trained using blog posts labeled with mood. Both unsupervised and supervised approaches were used. This paper compared results of many corpus based and knowledge based techniques which were performed on a big data set of 1000 headlines to find out the method that works best for emotion annotation.[9]

The purpose of this paper was to recognize emotions such as happiness, sadness and anger from sources containing text like e-mails and forums. Since emotion pairs like love and joy, fear and sadness have a close relationship among one another, only three emotions were selected for this study so that the emotions are clearly distinct from each other. Two different methods which are semantic network and keyword spotting method were analyzed and it was concluded that semantic emotion recognition engine was better at detecting emotions since there was no dependency on emotion keywords. This study only concentrates on texts regarding couples’ break up. Similar approach can be used for detecting emotion from texts regarding domestic violence, job dissatisfaction and effect of teenage problems on studies. The method of emotion detection used in this paper cannot detect emotions from technical or scientific texts.[10]

3. Proposed Methodology

The proposed method is to use the Django framework to implement a web based GUI which takes a speech dialogue as an input from the user in the form of an spoken audio recording then stores it into the MySQL database. Then using the IBM Watson Speech to Text API, the speech recording is converted to text. There is a limit of 20 words on the number of words which can be present in the text generated from speech to increase the efficiency of detecting the emotion. When the text has been generated then it is preprocessed using the NLP techniques. After the preprocessing step the text is then passed to the trained machine learning model for predicting the emotion for the text. As the framework is based on python it is suitable for implementing machine learning algorithms. The text obtained from
recording is acquired from the database and fed to the trained model for predicting the emotion. The predicted emotion is then displayed to the user with the help of the GUI.

Figure 1: Scheme

4. Implementation and results

The implementation of the project can be divided into two parts - front-end and back-end. The front-end is responsible for showing the web based GUI of the project and back-end is responsible for processing the data received through GUI and send the appropriate response for displaying it with the help of GUI to the user. Front-end displays the GUI which is created with the technologies HTML, CSS and JavaScript and rendered in the web browser with the help of Django framework. Back-end is used to get the data from front-end then process it inside different Python functions designated for different tasks then store the results in the MySQL database and then return the response back to front-end so that it can be displayed via the GUI.

The front-end is divided into two webpages - Home and Recordings. The Home webpage is further divided into three sections Recorder, Speech to Text and Find Emotion. Home is used to pass input data to the back-end server for processing and display the response sent by the server. The webpage Recordings is used to only display the information stored in the database in the form of a table. The table has four columns i.e., S.No. displays the values of id from the database, Speech Recording shows the name of the audio recordings stored in the database, Speech Text shows the text obtained from audio recordings and Emotion Detected displays the emotion predicted for the audio recordings using the values from Speech Text column.

In the Home webpage, Recorder section is used to record the spoken audio by capturing the audio stream from the mic in the WebM audio format when the user clicks on the start button. When the user
clicks on the stop button then the recording is stopped and is labeled with a unique name formed by combining current date and time and gets passed to its designated Python function in the Django back-end server which saves the recording’s name and the spoken audio in two separate columns in the MySQL database.

In the Speech to Text section when the convert button is clicked by the user then the current name of the recording is passed with the help of POST request to its designated Python function in the back-end server to fetch recording with the matching name from the database. The acquired recording is then handled by the IBM Watson Speech to Text API to convert the spoken audio in the recording to textual transcript. The API is based on deep learning technologies. The API returns the response in JSON format from which relevant data is extracted and converted to string in the same Python function and then passed to front-end to be displayed in the text box and also it is stored in the database for further processing to detect the emotion.

In the Find Emotion section when the user clicks on the find button then once again the current name of the recording is passed to its designated Python function in the back-end server with the help of POST request and the function searches for the matching name in the database and extracts the text associated to that recording name from the database. The acquired text then goes through the NLP preprocessing steps like making all the letters lowercase, removing punctuation and symbols, removing stop words using NLTK corpus, performing lemmatization and correcting letter repetitions. After preprocessing, in the feature extraction step count vector parameters are extracted from the text using the CountVectorizer function from the scikit-learn library. The idea here is to transform the text into an array with count of each word’s appearances. Then the Random Forest Classifier model is used to predict the emotion for the result obtained from the feature extraction step and then it is passed to the front-end for displaying the emotion in the text box to the user and also it is stored in the database for future reference.

In the training of the machine learning algorithms, the same preprocessing steps as above were performed on the data extracted from CrowdFlower emotion dataset. Then the target emotion labels i.e., aggression, anger, anticipation, awe, contempt, disapproval, disgust, fear, joy, love, neutral, optimism, remorse, sadness, submission, surprise and trust were encoded using the label encoder. And then the same feature extraction step as above was performed on the independent variable column i.e., sentences. The encoding and feature extraction steps are necessary as machine learning models are mathematical models and can only work on numerical data. Three machine learning algorithms were trained on the data extracted from the emotion dataset. The algorithms which were used are SVC (Support Vector Classification), Random Forest Classifier and Multinomial Naïve Bayes.

5. Results and discussion

The topic discussed in this paper comes under the category of multiclass or multinomial classification. In a multiclass classification problem, there are more than two classes available for the classification task. Each sample can be assigned only one class label out of all the class labels available. In our case the speech dialogue has to be assigned a label out of 17 class labels which itself is a very challenging task if the dataset is imbalanced i.e., when the classes are not distributed equally for the samples.
Figure 2: Bar graph for frequency of emotions in the training data

The emotion dataset from CrowdFlower which was used in this study contained only 2550 sentences. From the bar graph, it is evident that all the emotions are not equally divided among the sentences so it was not sufficient to train the algorithms for classifying every speech dialogue into a class label accurately. After training, the algorithms were tested on 50 new sentences in which the all the three algorithms mostly classified the sentences into the ‘neutral’ class label. This happened due to the reason that ‘neutral’ class label was the most used emotion in the training dataset. The metric that was used to compare the performance of the algorithms was accuracy score. The scores of all the three algorithms were below par but Random Forest Classifier performed slightly better than the Multinomial Naïve Bayes and Support Vector Classification algorithms. The algorithms were also trained by taking the values from emotion confidence column of the dataset into account but the testing yielded results which were too low for comparison so that method has not been included in this study.

| Algorithm            | Count Vectors | Accuracy Score (%) |
|----------------------|---------------|--------------------|
| Multinomial Naïve Bayes | YES           | ~ 25               |
Random Forest Classifier | YES | ~ 30
Support Vector Classification | YES | ~ 20

6. Conclusions

The results obtained as emotions for different speech dialogues were unsatisfactory due to the imbalanced dataset used for training the algorithms. For further work on this topic it is recommended that large annotated datasets be used for training an algorithm since to accurately predict the emotion for a text it is necessary that the algorithm learns from a large amount of emotion data. Considering the large amount of information for learning neural network algorithms are recommended as they will be the best choice for this kind of task. Detecting emotions is always being the most curious event of interaction, so get it extracted from the delivery would definitely pull a major area of concern. Many more elaborative fields can be judged and embarked on the basis of these parameters and findings.

References

[1] Jonathan Herzig, Michal Shmueli-Scheuer, David Konopnicki (2017) Emotion Detection from Text via Ensemble Classification Using Word Embeddings. ICTIR '17 Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval Pages 269-272

[2] Mary Jane C. Samonte, Richard Julian Paul G. Santiago, Hector Irvin B. Punzalan, Peter Joshua L. Linchangco (2017) Emotion Detection in Blog Posts Using Keyword Spotting and Semantic Analysis. ICCIP '17 Proceedings of the 3rd International Conference on Communication and Information Processing Pages 6-13

[3] Shiv Naresh Shivhare, Shakun Garg, Anitesh Mishra (2015) EmotionFinder: Detecting Emotion From Blogs and Textual Documents. International Conference on Computing, Communication & Automation DOI: 10.1109/CCAA.2015.7148343

[4] Ameeta Agrawal, Aijun An (2012) Unsupervised Emotion Detection from Text Using Semantic and Syntactic Relations. IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology DOI: 10.1109/WI-IAT.2012.170

[5] Linda Barros, Pilar Rodriguez, Alvaro Ortigosa (2012) Emotion Recognition in Texts for User Model Augmenting. INTERACCIÓN '12 Proceedings of the 13th International Conference on Interacción Persona-Ordenador Article No. 45

[6] Gábor Szucs, Gábor Magyar (2011) Classification, Recognition and Feedback in Text based Metacommunication. 2nd International Conference on Cognitive Infocommunications (CogInfoCom)

[7] Haji Binali, Chen Wu, Vidyasagar Potdar (2010) Computational Approaches for Emotion Detection in Text. 4th IEEE International Conference on Digital Ecosystems and Technologies
DOI: 10.1109/DEST.2010.5610650

[8] Alastair J. Gill, Robert M. French, Darren Gergle, Jon Oberlander (2008) The Language of Emotion in Short Blog Texts. CSCW '08 Proceedings of the 2008 ACM conference on Computer supported cooperative work Pages 299-30

[9] Carlo Strapparava, Rada Mihalcea (2008) Learning to Identify Emotions in Text. SAC '08 Proceedings of the 2008 ACM symposium on Applied computing Pages 1556-1560

[10] Huan Su Ling, Ranaivo Bali, Rosalina Abdul Salam (2006) Emotion detection using Keywords Spotting and Semantic Network. International Conference on Computing & Informatics. DOI: 10.1109/ICOCI.2006.5276495