|  |  |
| --- | --- |
| Internship Project Title | RIO-45: Automate detection of different emotions from textual comments and feedback |
| Project Title | Automate Detection of different emotions from textual comments and feedback |
| Name of the Company | TCS iON |
| Name of the Industry Mentor | Mr. Debashis Roy |
| Name of the Institute | Institute of Engineering & Management Kolkata |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Start Date | End Date | Total Effort (hrs.) | Project Environment | Tools used |
| 4 July 2020 | 2 Sept 2020 | 45 + | Jupyter Notebook | PyTorch , Deep Neural Network using GloVe Word Embeddings, LSTMs |

**Project Synopsis**

**Title of the project**

Automate Detection of different emotions from textual comments and feedback

**Domain**

Artificial Intelligence (AI)/ Machine Learning, Deep Learning

**Objective/Aim**

Social media is growing as a communication medium where people can express online their feelings and opinions on a variety of topics in ways they rarely do in person. Detecting sentiments and emotions in text have gained considerable amount of attention in the last few years.

**Technical Details**

We use word and document embeddings and a set of semantic features and apply CNN-LSTM and a fully connected neural network architectures to obtain performance results

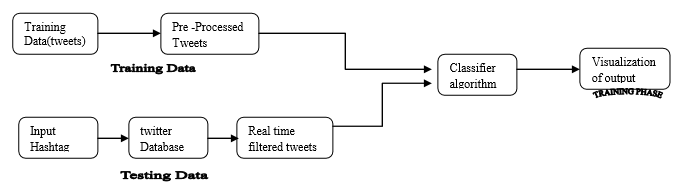
This study proposes a long-short term memory (LSTM)-based approach to text emotion recognition based on semantic word vector and emotional word vector of the input text. For each word in an input text, the semantic word vector is extracted from the word2vec model. Besides, each lexical word is projected to all the emotional words defined in an affective lexicon to derive an emotional word vector. An autoencoder is then adopted to obtain the bottleneck features from the emotional word vector for dimensionality reduction. The autoencoder bottleneck features are then concatenated with the features in the semantic word vector to form the final textual features for emotion recognition.

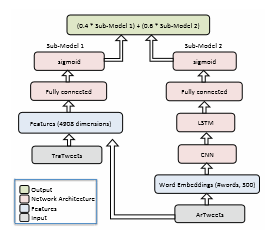
**Innovativeness & Usefulness**

Emotion Detection and Recognition from text is a recent field of research that is closely related to Sentiment Analysis. Sentiment Analysis aims to detect positive, neutral, or negative feelings from text, whereas Emotion Analysis aims to detect and recognize types of feelings through the expression of texts, such as anger, disgust, fear, happiness, sadness, and surprise. Emotion detection may have useful applications, such as:

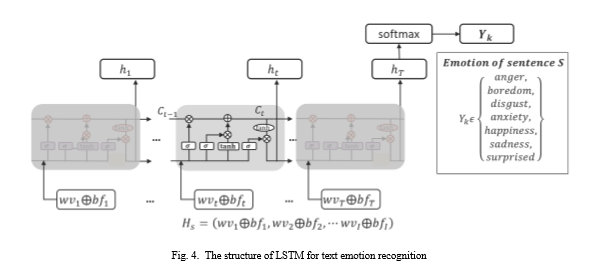
Gauging how happy our citizens are. Different indexes have different definitions; most evolve around economic, environmental, health, and social factors. Since the mid-2000s, Government and organizations around the world are paying increasing attention to the [happiness index](https://en.wikipedia.org/wiki/Happiness_economics).

**Solution Approach**





For the construction of the text emotion recognition model, the long-short term memory (LSTM) is applied in this study. LSTM is an extension of the recurrent neural network (RNN), which improves the learning ability for long-time sequence data and is suitable for learning the time-dependent word sequence used in this study. The structure of LSTM is briefly depicted in Fig



For the features, the CNN-based and LSTM-based models using the proposed features outperformed the methods simply using the semantic word vector or the bottleneck features of the emotional word vector. The proposed features could improve the performance because it considered both semantic word vector and the emotional word vector. For the structure of the emotion recognition model, the LSTM-based model performed better than the CNN-based model.

**Assumptions**

Detecting emotional state of a person by analysing a text document written by him/her appear challenging but also essential many times due to the fact that most of the times textual expressions are not only direct using emotion words but also result from the interpretation of the meaning of concepts and interaction of concepts which are described in the text document.

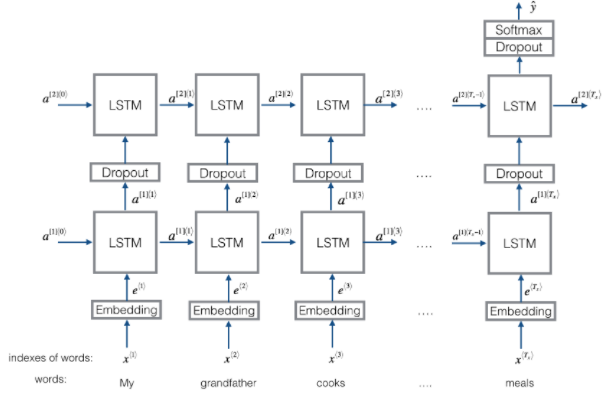
There are 6 emotion categories that are widely used to describe humans’ basic emotions, based on facial expression:  *anger, disgust, fear, happiness, sadness* and *surprise*. These are mainly associated with negative sentiment, with “Surprise” being the most ambiguous, as it can be associated with either positive or negative feelings. Interestingly, the number of basic human emotions has been recently “reduced”, or rather re-categorized, to just 4; *happiness, sadness, fear/surprise*, and *anger/disgust* . It is surprising to many that we only have 4 basic emotions. For the sake of simplicity for this code story, we will use the more widely-used 6 emotions. The question remains, however, how much of an emotion we can convey via text? This is especially interesting since facial expression and voice intonation convey over 70% of the intended feelings in spoken language.

Other from this I have done my detection of different emotions from textual comments on different emotions Loving , Playful , Happy , Annoyed , Foodie

**Algorithms**

This project is about performing emotion detection from text using PyTorch. For this project, we implemented an NLP task of creating a model to detect the emotion from text. We developed this using the PyTorch library where we created our Deep Neural Network using GloVe Word Embeddings, LSTMs and fully connected layers.

We will build an LSTM model that takes as input word sequences that will take word ordering into account. We will use 50-dimensional GloVe pre-trained word embeddings to represent words. We will then feed those as an input into an LSTM that will predict the most appropiate emotion for the text.

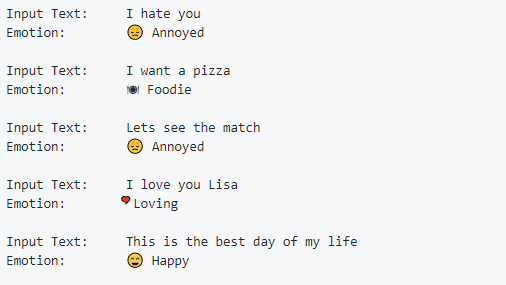


**Outcome**

We are able to successfully create a model that could achieve an accuracy of 84.4% in our training set and also were able to use our model on user inputted sentences and were able to get expected emotion outputs.

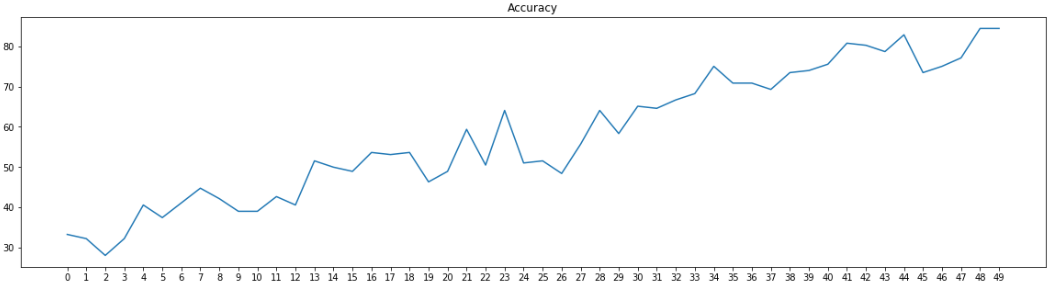
Test Loss: 1.064.. Test Accuracy: 0.844

The following results are emotion predictions on user inputted sentences



After training and testing the Model the output of the model comes in this way





**Application**

Emotions analysis of texts can be extended to any review related website for example product review to understand products popularity , movie review etc . It can also be highly useful in sub component technology such as detecting antagonistic ,heated language in mails ,context sensitive information detection ,spam detection etc . Determining consumer attitudes and trends is one of the major applications of emotional analysis of data.

**Conclusion/** **Enhancement Scope**

This study proposes an LSTM-based approach to detecting the user’s emotion expression in text. For the text input, the semantic word vectors were extracted from the lexical words by using word2vec method. An affective lexicon is adopted to project the lexical words to the emotion space to extract the emotional word vector. The bottleneck features are extracted from the emotional word vector using an autoencoder for dimensionality reduction. The bottleneck feature vector is then concatenated with the semantic word vector to construct the textual feature vector. Finally, the LSTM is used to characterize the temporal evolution of the emotion in each text input.

From the results, the proposed features, integrating semantic word vector with emotional word vector, performed better than the individual feature and reached an accuracy of 84.4% for text emotion recognition. In the future, more choices for data collection can be considered. Furthermore, some user-dependent information such as the personality of the user can also be taken into account when constructing the models of the system in order to make the system more personalized.

**Appendix**

Python: Python is very famous open source high level and dynamic programming language. There are many versions of it are already present to use. Version 2.7 is very famous and still many are stick on this version. For analytics version 3.0 or higher are suggested to use so that all types of APIs which may be used while classifying or performing any analytics can be easily combine and work.

LSTM: Long short-term memory is an artificial recurrent neural network architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It can not only process single data points, but also entire sequences of data.

Word2Vec: **Word2vec** is a technique for [natural language processing](https://en.wikipedia.org/wiki/Natural_language_processing). The word2vec algorithm uses a [neural network](https://en.wikipedia.org/wiki/Neural_network) model to learn word associations from a large corpus of text. Once trained, such a model can detect synonymous words or suggest additional words for a partial sentence. As the name implies, word2vec represents each distinct word with a particular list of numbers called a [vector](https://en.wikipedia.org/wiki/Vector_(geometry)). The vectors are chosen carefully such that a simple mathematical function (the [cosine similarity](https://en.wikipedia.org/wiki/Cosine_similarity) between the vectors) indicates the level of [semantic similarity](https://en.wikipedia.org/wiki/Semantic_similarity) between the words represented by those vectors.

**Link to Code and Executable File**

<https://github.com/aman7401/Emotion-detection-from-text-using-PyTorch-and-Federated-Learning/blob/master/Emotion_detection_from_text_using_PyTorch.ipynb>

**REFERENCES**

[1] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, “Sentiment analysis of twitter data,” in Proceedings of the workshop on languages in social media. Association for Computational Linguistics, 2011, pp. 30–38.

[2] P. Ekman, “Facial expression and emotion.” American psychologist, vol. 48, no. 4, p. 384, 1993.

[3] F. Harrag, “Estimating the sentiment of arabic social media contents: A survey,” in 5th International Conference on Arabic Language Processing, 2014.

[4] M. Abdullah and M. Hadzikadic, “Sentiment analysis on arabic tweets: Challenges to dissecting the language,” in International Conference on Social Computing and Social Media. Springer, 2017, pp. 191–202.

[5] K. Shaalan, “Rule-based approach in arabic natural language processing,” The International Journal on Information and Communication Technologies (IJICT), vol. 3, no. 3, pp. 11– 19, 2010.

[6] Z. Eviatar and R. Ibrahim, “Why is it hard to read arabic?” in Handbook of Arabic literacy. Springer, 2014, pp. 77–96.

[7] C. Guggilla, T. Miller, and I. Gurevych, “Cnn-and lstm-based claim classiﬁcation in online user comments,” in Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, 2016, pp. 2740–2751.

[8] Y. Kim, “Convolutional neural networks for sentence classiﬁcation,” arXiv preprint arXiv:1408.5882, 2014.

[9] S. M. Mohammad and S. Kiritchenko, “Understanding emotions: A dataset of tweets to study interactions between affect categories,” in Proceedings of the 11th Edition of the Language Resources and Evaluation Conference, Miyazaki, Japan, 2018.

[10] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efﬁcient estimation of word representations in vector space,” arXiv preprint arXiv:1301.3781, 2013.

[11] S. M. Mohammad and F. Bravo-Marquez, “Wassa2017 shared task on emotion intensity,” arXiv preprint arXiv:1708.03700, 2017.

[12] F. Bravo-Marquez, M. Mendoza, and B. Poblete, “Meta-level sentiment models for big social data analysis,” KnowledgeBased Systems, vol. 69, pp. 86–99, 2014.

[13] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradientbased learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[15] R. Plutchik, “Emotions and psychotherapy: A psychoevolutionary perspective,” in Emotion, psychopathology, and psychotherapy. Elsevier, 1990, pp. 3–41.

**Links**

<https://medium.com/@sabber/classifying-yelp-review-comments-using-lstm-and-word-embeddings-part-1-eb2275e4066b#:~:text=Build%20a%20neural%20network%20with%20LSTM,-In%20the%20following&text=The%20network%20starts%20with%20an,word%20in%20a%20meaningful%20way.>

<https://github.com/krishnaik06/Word-Embedding/blob/master/Untitled2.ipynb>

<https://github.com/krishnaik06/Natural-Language-Processing/blob/master/Toeknization.py>

<https://github.com/krishnaik06/Natural-Language-Processing/blob/master/Toeknization.py>

<https://ieeexplore.ieee.org/abstract/document/8614159>

<https://www.youtube.com/results?search_query=LSTm>

<https://github.com/krishnaik06/Word-Embedding/blob/master/Untitled2.ipynb>

<https://www.analyticsvidhya.com/blog/2020/03/pretrained-word-embeddings-nlp/>

<https://medium.com/@sabber/classifying-yelp-review-comments-using-cnn-lstm-and-visualize-word-embeddings-part-2-ca137a42a97d>

<https://medium.com/@sabber/classifying-yelp-review-comments-using-lstm-and-word-embeddings-part-1-eb2275e4066b#:~:text=Build%20a%20neural%20network%20with%20LSTM,-In%20the%20following&text=The%20network%20starts%20with%20an,word%20in%20a%20meaningful%20way.>

<https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec-652d0c2060fa>

<https://towardsdatascience.com/light-on-math-ml-intuitive-guide-to-understanding-glove-embeddings-b13b4f19c010>

<https://github.com/krishnaik06/Natural-Language-Processing/blob/master/Toeknization.py>