

Analysis of Twitter Data Using Deep Learning Approach: LSTM

Vikas Malik

Department of Computer Science
Manav Institute of Technology & Management
Hisar, India
e-mail: vikasprof.om@gmail.com

Amit Kumar

Department of Computer Science
Manav Institute of Technology & Management
Hisar, India
e-mail: amitkrnayak10@gmail.com

Abstract— Sentiment analysis the procedure of computationally identifying and categorizing evaluations expressed in a chunk of text, especially with a view to decide whether the writer’s mind-set toward a selected subject matter, product, etc. is high-quality, poor, or impartial[1]. Now a days the growth of social websites, running a blog offerings and electronic media con-tributes big amount of consumer supply messages which includes customer reviews, remarks and evaluations. Sentiment evaluation is an important term cited gather facts in a source with the aid of the usage of NLP, computational[2] linguistics and text analysis and to make decision through subjective information extracting and analyzing opinion, figuring out advantageous and bad opinions measuring how definitely and negatively an entity (public ,organization, product) is concerned.

in the beyond decade , researcher have performed the sentiment analysis using device getting to know techniques which include guide vector gadget, naive bayes , maximum entropy method etc. Sentient analysis on social media textual content received lot of recognition because it includes pointers and pointers. lately deep gaining knowledge of methods like long short-term memory (LSTM) and convolution neural network (CNN) have gained recognition by means of displaying promising effects for speech and photograph processing, obligations in NLP through learning functions wealthy deep illustration from the facts robotically.

Keywords- Twitter, Sentiment Analysis (SA), Machine Learning, Naïve Bayes, Deep Learning, Natural Language Processing (NLP), Long Short Term Memory (LSTM), Recurrent Neural Networks (RNN).

I. LSTM INTRODUCTION

Long short-term memory (LSTM) gadgets (or blocks) are a constructing unit [3] for layers of a recurrent neural network (RNN). A RNN composed of LSTM units is often known as an LSTM network. A common LSTM unit consists of a cellular, an enter gate, an output gate and a overlook gate. The cell is liable for ”remembering” values over arbitrary time periods; therefore the phrase ”memory” in LSTM. each of the 3 gates can be idea of as a ”traditional” synthetic neuron, as in a multi-layer (or feed-ahead) neural network: that is, they compute an activation (the usage of an activation characteristic) of a weighted sum. Intuitively, they may be thought as regulators of the flow of values that goes via the connections of the LSTM; therefore the denotation”gate”. There are connections[4] among these gates and the cell.

The expression long brief-term refers back to the reality that LSTM is a model for the short-term reminiscence that could closing for an extended time frame. An LSTM is nicely-proper to classify procedure and are expecting time collection given time lags of unknown length and length among important occasions. LSTM have been advanced to cope with the exploding and vanishing gradient trouble whilst training traditional RNNs[5]. Relative insensitivity to hole duration offers an advantage to LSTM over opportunity RNNs hidden Markov models and other sequence learning methods in numerous applications.

II. SENTIMENT ANALYSIS USING LSTM

Here we use the example of evaluations to expect sentiment despite the fact that it may be carried out more generically to other domain names for example sentiment evaluation for tweets, comments, customer remarks, and so forth. Complete concept here is that movie critiques are fabricated from sequence of phrases and orders of phrases encode lot of information this is beneficial to predict sentiment.

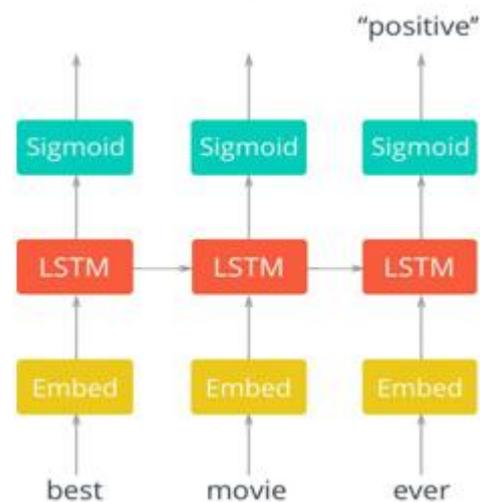


Fig:-1 LSTM Network

Step 1is to map words to word embeddings Step 2 is the RNN that gets a sequence of vectors as enter and considers the order of the vectors to generate prediction. Right here, nicely pass in

words to an embedding layer. you may virtually educate up an embedding with word2vec and use it here. but it's excellent sufficient to just have an embedding layer and let the network research the embedding desk on its very own.[6]

From the embedding layer, the brand new representations may be exceeded to LSTM cells. Those will add recurrent connections to the community so we are able to include records approximately the series of words in the records. Subsequently, the LSTM cells will visit a sigmoid output layer right here. we are the usage of the sigmoid due to the fact we're looking to expect if this article has high-quality or bad sentiment. The output layer will just be a single unit then, with a sigmoid activation feature[7,8].

We don't care approximately the sigmoid outputs except for the very last one, we can ignore the relaxation. properly calculate the cost from the output of the remaining step and the schooling label.[9]

A. About Data Set

This data originally came from Crowdfower Data for everyone library.

<http://www.crowdfower.com/data-for-everyone>.

As the original source says,

We looked via tens of lots of tweets about the early August GOP debate in Ohio and asked contributors to do both sentiment evaluation and information categorization. members had been asked if the tweet turned into relevant, which candidate turned into noted, what concern became noted, after which what the sentiment changed into for a given tweet. We've removed the non-applicable messages from the uploaded dataset.

I decided to only do sentiment analysis on this dataset, therefore I dropped the unnecessary columns, keeping only sentiment and text.

B. Processing Data Set

A tweet contains a lot of opinions about the data which are expressed in special methods via one-of-a-kind users. The twitter dataset used on this survey paintings is already classified into classes viz. bad and fantastic polarity and thus the sentiment evaluation of the records will become clean to observe the impact of numerous capabilities. The raw information having polarity is relatively liable to inconsistency and redundancy. Preprocessing of tweet consist of following points-

- Remove all URLs (e.g. www.xyz .com), hash tags (e.g. topic), targets (@u ser-name).
- Correct the spellings; sequence of repeated

characters is to be handled .

- Replace all the emoticons with their sentiment.
- Remove all punctuations, symbols, numbers .
- Expand Acronyms (Use a acronym dictionary).
- Remove Non - English Tweets.

Stop Word: Stop Words are phrases which do now not include critical significance for use in seek Queries. Usually these words are filtered out from seek queries because they return full-size quantity of useless facts (the, for, this and many others.).

For the use of our Machine learning algorithms we first need to transform words to vectors. After that we will run our algorithms on that characteristic vector. We then feed this vectorised statistics to our algorithm.[10]

I am dropping the 'Neutral' sentiments as my goal was to only differentiate positive and negative tweets. After that, I am filtering the tweets so only valid texts and words remain. Then, I define the number of max features as 2000 and use Tokenizer to vectorize and convert text into Sequences so the Network can deal with it as input.

{#code for processing data}

```
data = pd.read_csv('../input/Sentiment.csv')
# Keeping only the necessary columns
data = data['text', 'sentiment']
data = data[data.sentiment != 'Neutral']
data['text'] = data['text'].apply(lambda x: x.lower())
data['text'] = data['text'].apply(lambda x: re.sub
('^a-zA-z0-9\s', '' , x))
print(data[data['sentiment'] == 'Positive'].size)
print(data[ data['sentiment'] == 'Negative'].size)
for idx, row in data.iterrows():
row[0] = row[0].replace('rt', '\n')
max_features = 2000 tokenizer = Tokenizer
(nb words=max_features, split='\n')
tokenizer.fit_on_texts(data['text'].values)
X = tokenizer.texts_to_sequences
(data ['text']. values)
X = pad_sequences(X)
```

III. TENSORFLOW

C. About TensorFlow

TensorFlow is an open source software library for high performance numerical computation. Its flexible architecture allows easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices. Originally developed by researchers and engineers from the Google Brain team within Google's AI organization, it comes with strong support for machine learning and deep learning and the flexible

numerical computation core is used across many other scientific domains.[19]

TensorFlow is an interface for expressing machine getting to know algorithms, and an implementation for executing such algorithms. A computation expressed the use of TensorFlow can be completed with little or no exchange on a extensive form of heterogeneous systems, starting from cell gadgets which includes phones and capsules as much as huge-scale disbursed structures of masses of machines and lots of computational devices inclusive of GPU playing cards. The system is flexible and can be used to express a wide variety of algorithms, including training and inference algorithms for deep neural network models, and it has been used for conducting research and for deploying machine learning systems into production across more than a dozen areas of computer science and other fields, including speech recognition, computer vision, robotics, information retrieval, natural language processing, geographic information extraction, and computational drug discovery. This paper describes the TensorFlow interface and an implementation of that interface that we have built at Google. The TensorFlow API and a reference implementation were released as an open-source pack-age under the Apache 2.0 license in November, 2015 and are available at www.tensorflow.org.[19]

IV. ANALYSIS USING LSTM

An LSTM network has been composed. Note that 'embed_dim, lstm_out, batch_size, dropout_x variables are hyper parameters, their values are somehow intuitive, can be and must be played with in order to achieve good results. Please also note that I am using softmax as activation function. The reason is that our Network is using categorical crossentropy, and softmax is just the right activation method for that.

```

embed_dim = 128
lstm_out = 196
model=Sequential()
model.add(Embedding(max_features,embed_dim,
input_length=X.shape[1],dropout=0.2))
model.add(LSTM(lstm_out, dropout U=0.2,
dropout_W=0.2))
model.add(Dense(2,activation='softmax'))
model.compile(loss='categorical_crossentropy',
optimizer='adam',metrics=['accuracy'])
print(model.summary())
    
```

V. UNDERSTANDING LSTM NETWORK

To apprehend LSTM we need to apprehend RNNs. A recurrent neural network (RNN) is a class of artificial neural network where connections among devices shape a directed

graph along a series. This allows it to exhibit dynamic temporal behavior for a time sequence.

A. Recurrent Neural Network

Humans don't begin their questioning from scratch every second. As you examine this essay, you understand every phrase primarily based on your know-how of preceding words. You don't throw the entirety away and start questioning from scratch again. Your mind have persistence traditional neural networks can't try this, and it seems like a primary shortcoming. as an example, consider you need to classify what kind of event is taking place at each point in a movie. It's doubtful how a traditional neural community may want to use its reasoning approximately previous events inside the film to inform later ones.

Recurrent neural networks address this difficulty. They're networks with loops in them, permitting statistics to persist.

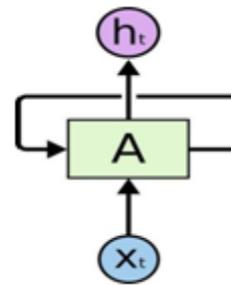


Fig:-2 Recurrent Neural Networks having loops

In the above diagram, a bit of neural network, A, appears at some input x_t and outputs h_t . A loop lets in facts to be handed from one step of the network to the subsequent[11].

These loops make recurrent neural networks seem type of mysterious. but, in case you suppose a chunk extra, it turns out that they aren't all that extraordinary than a normal neural network. A recurrent neural network may be concept of as a couple of copies of the same community, every passing a message to a successor. don't forget what happens if we unroll the loop[12].

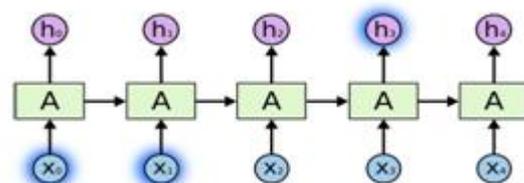


Fig:-3 An unrolled recurrent neural network

This chain-like nature reveals that recurrent neural networks are intimately related to sequences and lists. They're the natural architecture of neural network to use for such data.

B. The Problem Of Long Term Dependencies

One of the appeals of RNNs is the concept that they are probably able to connect previous data to the prevailing undertaking, consisting of the usage of preceding video frames may inform the knowledge of the prevailing frame. If RNNs should do this, they'd be extraordinarily beneficial. but can they? It depends.

Occasionally, we only need to study latest statistics to carry out the prevailing project. For example, recall a language model seeking to predict the following word primarily based at the previous ones. If we're looking to predict the closing phrase inside the clouds are in the sky, we don't need any similarly context it's quite apparent the subsequent phrase is going to be sky. In such cases, where the space between the applicable facts and the place that it's wanted is small, RNNs can learn how to use the beyond data.

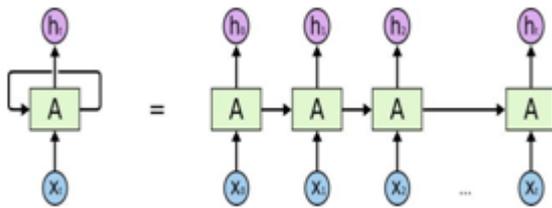


Fig 4: Recurrent Neural Network

But there also are instances wherein we need extra context. remember trying to predict the closing phrase inside the textual content "I grew up in France I communicate fluent French". recent information suggests that the subsequent word is probably the name of a language, but if we need to narrow down which language, we need the context of France, from in addition back. It's entirely viable for the gap between the relevant information and the factor in which it's far had to come to be very massive

C. LSTM Network

Long Short Term Memory networks simply called LSTMs are a special form of RNN, capable of learning lengthy-time period dependencies. They had been brought by means of Hochreiter Schmidhuber (1997), and had been subtle and popularized through many people in following work. They paintings quite properly on a big form of troubles, and are now extensively used[13].

LSTMs are explicitly designed to avoid the lengthy-time period dependency problem. Remembering facts for lengthy periods of time is almost their default behavior, no longer something they conflict to study.[14]

All recurrent neural networks have the shape of a sequence of repeating modules of neural network. In fashionable RNNs, this repeating module will have a very easy structure, such as a unmarried tanh layer.

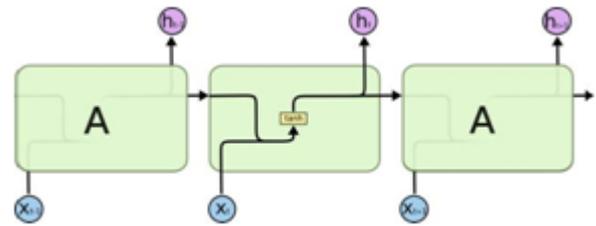


Fig:-5 Repeating module in a standard contains a single layer.

LSTMs also have this chain like shape, but the repeating module has a exceptional shape. rather of getting a unmarried neural network layer there are 4 layers interacting in a totally unique way.

D. The Core Idea Behind LSTMs

The key to LSTMs is the cell kingdom, the horizontal line jogging thru the top of the diagram. The mobile country is kind of like a conveyor belt. It runs directly down the whole chain, with only some minor linear interactions. Its very clean for records to simply go with the flow alongside it unchanged.

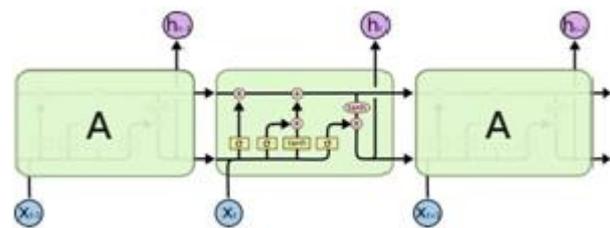


Fig 6:-Repeating module in an LSTM contains four interacting layers.

The LSTM does have the capability to get rid of or add records to the cell state, carefully regulated via systems referred to as gates. The sigmoid layer outputs numbers among zero and one, describing how a good deal of each factor ought to be allow through. A cost of zero approach permit not anything through, even as a value of 1 means permit everything thru.

Gates are a way to optionally permit records through. they are composed out of a sigmoid neural internet layer and a factor sensible multiplication operation.

Compact form of the equations for the ahead skip of an LSTM unit with a neglect gate.

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \tag{1}$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \tag{2}$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \tag{3}$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \quad (4)$$

$$h_t = o_t \circ \sigma_h(c_t) \quad (5)$$

VI. RESULT

The performance of sentiment classification can be evaluated by using four indexes calculated as the following equations:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})$$

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

$$F1 = (2 * \text{Precision} * \text{Recall}) / (\text{Precision} + \text{Recall})$$

In which TP, FN, FP and TN refer respectively to the number of true positive instances, the number of false negative instances, the number of false positive instances and the number of true negative instances.

TABLE I. ACTUAL POSTIVE AND ACTUAL NEGATIVE

	Predicted Positive	Predicted Negative
Actual Positive	TP	FN
Actual Negative	FP	TN

After analysing result of our machine learning model on testing set, we obtained following result:

TABLE II. CALCULATED TWEETS

True Positive (TP)	130
False Negative (FN)	100
False Positive (FP)	62
True Negative (TN)	787

Now we will Calculate (Accuracy, Precision, Recall and F-Score) of our result:

TABLE III. FINAL RESLUT

Accuracy	84.98
Precision	67.70
Recall	56.52
F-Score	61.60

Here we have achieved good result as compared to Naïve Bayes algorithm. Comparison of Naïve Bayes and LSTM is given below in the table and graph. [15,16,17].

TABLE IV. COMPARISON OF NAÏVE BAYES AND LSTM

RESULT SET	NAÏVE BAYES	LSTM
------------	-------------	------

Accuracy	81.64	84.98
Precision	60.6	67.70
Recall	39.56	56.52
F-Score	49.25	61.60

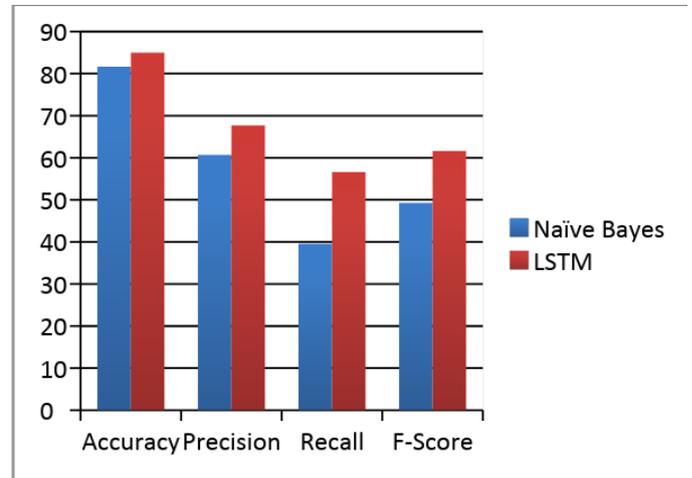


Fig 7:-Result comparison of Naïve Bayes and LSTM.

VII. APPLICATION OF SENTIMENT ANALYSIS

- Sentiment analysis has many programs and advantages on your enterprise and organization. it is able to be used to present your commercial enterprise valuable in-attractions into how human beings feel about your product brand or service.
- When carried out to social media channels, it could be used to become aware of spikes in sentiment, thereby permitting you to discover ability product advocates or social media influencers.
- It is able to be used to pick out while capability poor threads are emerging online concerning your business, thereby allowing you to be proactive in managing it greater quick.
- Sentiment analysis may also be carried out in your corporate community, for instance, by means of making use of it to your e mail server, emails can be monitored for their preferred tone. for example, Tone Detector is an Outlook upload-in that deter-mines the tone of your email as you type. Like an emotional spell checker for all your outgoing e-mails.

VIII. CONCLUSION

Finally measuring the range of correct guesses. it's far clear that finding negative tweets goes thoroughly for the community but identifying whether is tremendous isn't clearly. My knowledgeable guess here is that the fine training set is dramatically smaller than the poor, consequently the "awful" effects for high quality tweets. we can conclude that greater the purifier records, greater correct results can be obtained. we

are able to attention on the observe of mixing machine getting to know technique into opinion lexicon approach with a view to enhance the accuracy of sentiment classification and adaptive capacity to variety of domains and different languages.

IX. REFERENCES

- [1] Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1), 1-167.
- [2] Medhat, W., Hassan, A., Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5(4), 1093-1113.
- [3] Go, A., Bhayani, R., Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report*, Stanford, 1(12)
- [4] Btow, F., Schultze, F., Strauch, L. *Semantic Search: Sentiment Analysis with Machine Learning Algorithms on German News*.
- [5] Pak, A., Paroubek, P. (2010, May). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In *LREc* (Vol. 10, No. 2010).
- [6] Gokulakrishnan, B., Priyanthan, P., Ragavan, T., Prasath, N., Perera, A. (2012, December). Opinion mining and sentiment analysis on a twitter data stream. In *Advances in ICT for emerging regions (ICTer)*, 2012 International Conference on (pp. 182-188). IEEE.
- [7] Hallsmar, F., Palm, J. (2016). Multi-class sentiment classification on twitter using an emoji training heuristic.
- [8] Turney, P. D. (2002, July). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 417-424). Association for Computational Linguistics.
- [9] Park, S., Kim, Y. (2016, June). Building thesaurus lexicon using dictionary-based approach for sentiment classification. In *Software Engineering Research, Management and Applications (SERA)*, 2016 IEEE 14th International Conference on (pp. 39-44). IEEE.
- [10] Hatzivassiloglou, V., McKeown, K. R. (1997, July). Predicting the semantic orientation of adjectives. In *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics* (pp. 174-181). Association for Computational Linguistics.
- [11] Ding, X., Liu, B., Yu, P. S. (2008, February). A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining* (pp. 231-240). ACM.
- [12] Thakkar, H., Patel, D. (2015). Approaches for sentiment analysis on twitter: A state-of-art study. *arXiv preprint arXiv:1512.01043*.
- [13] Turney, P. D. (2002, July). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on association for computational linguistics* (pp. 417-424). Association for Computational Linguistics.
- [14] Rothfels, J., Tibshirani, J. (2010). Unsupervised sentiment classification of English movie reviews using automatic selection of positive and negative sentiment items. *CS224N-Final Project*.
- [15] Zagibalov, T., Carroll, J. (2008, August). Automatic seed word selection for unsupervised sentiment classification of Chinese text. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1* (pp. 1073-1080). Association for Computational Linguistics.
- [16] Tang, B., Kay, S., He, H. (2016). Toward optimal feature selection in naive Bayes for text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 28(9), 2508-2521.
- [17] Go, A., Bhayani, R., Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report*, Stanford, 1(12).
- [18] Gautam, G., Yadav, D. (2014, August). Sentiment analysis of twitter data using machine learning approaches and semantic analysis. In *Contemporary computing (IC3)*, 2014 seventh international conference on (pp. 437-442). IEEE.
- [19] TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, <https://www.tensorflow.org/>, Software available from tensorflow.org