Smart Reply Prediction using LSTM

SHIRISHA KAMPATI , HARIPRIYA MUPPIDI , S VIJAYA LAKSHMI

Abstract— Smart Reply Prediction using LSTM model is used to predict responses to the queries. This model generates semantically diverse suggestions that can be used as a complete text response.Stanford Question and Answer Dataset(SQuAD) from Kaggle, Question and Answer Datasets related to Music,Groceries and Video games from Github were considered here.Questions and their respective replies were the only selected columns from the dataset.Bidirectional Long Short-Term Memory model and Natural language processing steps were used here. Predicting correct answers to a question is estimated to be 60% using Micro F1-score. Since, this process is automated, it not only saves user's money for hiring a personal assistant but also saves a lot of time and turns out to be very performance efficient and productive.

Index Terms— F1-score, Bidirectional Long Short-Term Memory, Natural Language Processing , Batch Normalization.

1 INTRODUCTION

QUESTION answering is an important NLP task.It is always exciting when one seeks someone or some enterprise out via email or text and they respond instantly in no time.In today's society,especially as technology grows,people are looking for immediate response and satisfaction.Users spend a lot of time in reading,replying and organizing their replies.With the rapid increase in question-answer overload in texts/emails,it has become increasingly challenging for users to process and respond to incoming messages.

The need to query information content available in various formats including structured and unstructured data has become increasingly important. Thus, Question Answering Systems (QAS) are essential to satisfy this need. A question answering (QA) system is a system designed to answer questions posed in natural language. Some QA systems draw information from a source such as text or an image in order to answer a specific question. In an enterprise setting, they can be used for much more than chatbots and voice assistants. For example, smart algorithms can be trained to do the following:Administration,Customer service and Marketing. Obtaining possible replies to a specific query helps the user to respond to a client in less time.

2 RELATED WORKS

Much work exists on natural language dialogues in public domains such as Twitter, but it has largely focused on social media tasks like predicting whether or not a response a made [1], predicting next word only[2], or curating threads[3]. Full response prediction was initially attempted in [4], which approached the problem from the perspective of machine translation: given a Twitter post, "translate" it into a response using phrase-based statistical machine translation(SMT). The paper's approach is similar,but rather than using SMT we use the neural network machine translation model proposed in [5], called "sequence-to-sequence learning".

Sequence-to sequence learning, which makes use of long short-term memory networks (LSTMs) [6] to predict sequences of text,was originally applied to Machine Translation but has since seen success in other domains such as image captioning [7] and speech recognition[8].Other recent works have also applied recurrent neural networks (RNNs) or LSTMs to full response prediction [9], [10], [11], [12]. In [9] the authors rely on having an SMT system to generate n-best lists, while [11] and [12], like this work,develop fully generative models.Our approach is most similar to the Neural Conversation Model [12], which uses sequence-to-sequence learning to model tech support chats and movie subtitles.

A Question-Answer model can be pre-programmed with specific and detailed responses. It analyses and processes the content of each query and perform actions based on this information.Users can handle massive volumes of text/email at the speed of today's market demands. They get timely and personalized responses that they expect.They can also focus on what matters most – user relationships.

3 BIDIRECTIONAL LSTM

3.1 Analysis of Datasets

In this paper,Stanford Question and Answer Dataset(SQuAD) (.json from Kaggle), Music Dataset (.csv from Github), Video games Dataset (.csv from Github), Grocery Dataset (.csv from Github) and General Question Answer Dataset (.csv from Github) were loaded.

The following are the attributes in the datsets:

1.Questions : These are the general queries asked by users.
2.Answers : These are the replies generated to the specified questions.
3.DifficultyfromQuestioner : This is the difficulty level of the questions.
4.DifficultyfromAnswerer : This is the difficulty level of the generated answers.
5.ArticleTitle : This has the questions related to a specific title.
6.ArticleFile : This is the specified path of the Question based on the ArticleTitle.

Microsoft Excel - qa_dataset

	A	B	C	D	E	F
1	ArticleTitle	Question	Answer	DifficultyFromQuestioner	DifficultyFromAnswerer	ArticleFile
2	Alessandro_Volta	Was Volta an Italian physicist?	yes	easy	easy	data/set4/a10
3	Alessandro_Volta	Is Volta buried in the city of Pittsburgh?	no	easy	easy	data/set4/a10
4	Alessandro_Volta	Did Volta have a passion for the study of electricity?	yes	easy	medium	data/set4/a10
5	Alessandro_Volta	What is the battery made by Volta credited to be?	the first cell	medium	medium	data/set4/a10
6	Alessandro_Volta	What important electrical unit was named in honor of Volta?	the volt	medium	medium	data/set4/a10
7	Alessandro_Volta	What important electrical unit was named in honor of Volta?	volt	medium	medium	data/set4/a10
8	Alessandro_Volta	Where did Volta enter retirement?	Spain	medium	medium	data/set4/a10
9	Alessandro_Volta	Is it a disadvantage for something to be unsafe to handle?	yes	hard	too hard	data/set4/a10
10	Alessandro_Volta	Was Lombardy under Napoleon's rule in 1800?	yes	hard	hard	data/set4/a10
11	Alessandro_Volta	Was the Italian 10.000 lira banknote created before the euro?	yes	hard	hard	data/set4/a10
12	Alessandro_Volta	For how many years did Alessandro Volta live?	53	NULL	medium	data/set4/a10
13	Alessandro_Volta	Did Alessandro Volta live to be 80 years old?	no	NULL	easy	data/set4/a10
14	Alessandro_Volta	What was Alessandro Volta`s profession?	physisist	NULL	easy	data/set4/a10
15	Alessandro_Volta	How old was Alessandro Volta when he died?	82	NULL	hard	data/set4/a10
16	Alessandro_Volta	How many years ago was it when Volta married the daughter of Count Ludov	215	NULL	hard	data/set4/a10
17	Alessandro_Volta	Is the electrolyte sulphuric acid?	Yes	NULL	easy	data/set4/a10
18	Alessandro_Volta	Is volta buried in the city of Como?	Yes	NULL	easy	data/set4/a10
19	Alessandro_Volta	Was his 1800 paper written in French?	Yes	NULL	easy	data/set4/a10
20	Alessandro_Volta	Before 1796, was Lombardy ruled by Austria?	yes	NULL	medium	data/set4/a10
21	Alessandro_Volta	Did he receive the Society`s 1794 Copley Medal?	Yes	NULL	easy	data/set4/a10
22	Alessandro_Volta	Did he experiment with individual cells?	Yes	NULL	easy	data/set4/a10
23	Alessandro_Volta	When did lombardy come under Napoleon`s rule?	From 1796 to 1815	NULL	easy	data/set4/a10
24	Alessandro_Volta	Where did he publish his invention of the Voltaic pile battery?	the Philosophical Tra	NULL	medium	data/set4/a10
25	Alessandro_Volta	Did he become professor of experimental physics at the University of Pavia?	Yes	NULL	easy	data/set4/a10
26	Alessandro_Volta	Is it true that his passion been always the study of electricity?	Yes	NULL	easy	data/set4/a10
27	Alessandro_Volta	His passion been always the study of what?	Electricity	NULL	easy	data/set4/a10
28	Alessandro_Volta	Is it true that Volta married the daughter of Count Ludovico Peregrini?	Yes	NULL	easy	data/set4/a10
29	Alessandro_Volta	Is it true that he published his invention of the Voltaic pile battery?	Yes	NULL	easy	data/set4/a10
30	Amedeo_Avogadro	Was Amedeo Avogadro Italian?	Yes	easy	easy	data/set4/a8
31	Amedeo_Avogadro	Did Amedeo Avogadro graduate?	Yes	easy	easy	data/set4/a8
32	Amedeo_Avogadro	Did Avogadro live in England?	No	easy	easy	data/set4/a8
33	Amedeo_Avogadro	Where was Avogadro a professor of physics?	University of Turin	medium	medium	data/set4/a8

Fig 1 General Question and Answer Dataset

3.2 Data Preprocessing

Only Question and Answer attributes are considered here.Remaining attributes are considered as noisy data. From the datasets,we consider the stopwords and decontract them. Beautiful Soup is a Python library for pulling data out of HTML and XML files. The length of questions and replies, number of unique replies in the dataset and number of times the reply occurs in the whole set are counted. Repeated replies are plotted as graphs using matplotlib. The symbols such as : [';', ':', '!', "*",'.',')','(','?'] are removed from questions. Stemming is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form.It is applied on preprocessed questions.

Data is preprocessed in the following manner:converting into lower case, removing all special characters(stop words), removing data where answers are incorrect, trimming the data

Remove the unnecessary attributes from the dataset and combine all the datasets using pandas dataframes. After Data Cleaning, use this integrated dataset for next step.

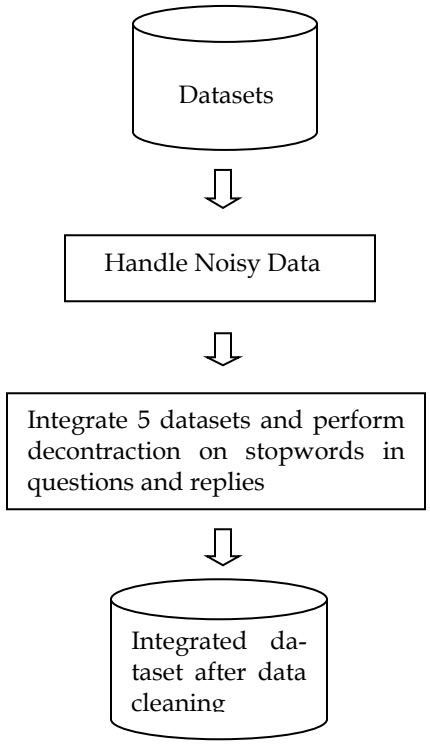


Fig 2 Data Preprocessing Technique

3.3 Long Short-Term Memory Algorithm

An Long Short Term Memory(LSTM) has a similar control flow as a recurrent neural network. It processes data passing on information as it propagates forward.

A typical LSTM network is comprised of different memory blocks called cells. There are two states that are being transferred to the next cell; the cell state and the hidden state. The memory blocks are responsible for remembering things and manipulations to this memory is done through three major mechanisms, called gates.

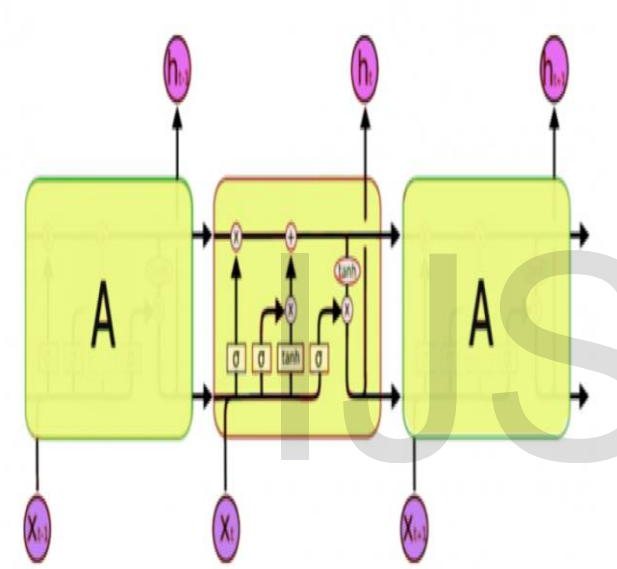


Fig 3 LSTM model (here activation function used is "relu" instead of "tanh")

There are three types of gates in LSTM.They are:

a) Forget gate
b) Input gate
c) Output gate

a) Forget gate :

A forget gate is responsible for removing information from the cell state. The information that is no longer required for the LSTM to understand things or the information that is of less importance is removed via multiplication of a filter.

b) Input gate :

The input gate is responsible for the addition of information to the cell state. This addition of information is basically three-step process as seen from the diagram above.

1. Regulating what values need to be added to the cell state by involving a sigmoid function. This is basically very similar to the forget gate and acts as a filter for all the information from h_t-1 and x_t.

2. Creating a vector containing all possible values that can be added (as perceived from h_t-1 and x_t) to the cell state. This is done using the relu function.

3. Multiplying the value of the regulatory filter (the sigmoid gate) to the created vector (the relu function) and then adding this useful information to the cell state via addition operation.

c) Output gate :

The job of selecting useful information from the current cell state and showing it out as an output is done via the output gate. The functioning of an output gate can again be broken down to three steps:

1. Creating a vector after applying relu function to the cell state.

2. Making a filter using the values of h_t-1 and x_t, such that it can regulate the values that need to be output from the vector created above. This filter again employs a sigmoid function.

3. Multiplying the value of this regulatory filter to the vector created in step 1, and sending it out as a output and also to the hidden state of the next cell.

3.4 TRAINING THE MODEL

3.4.1 Batch Normalization

Batch normalization is a technique designed to automatically standardize the inputs to a layer in a deep learning neural network. The BatchNormalization layer is added to the model to standardize raw input variables or the outputs of a hidden layer.Once implemented, batch normalization has the effect of dramatically accelerating the training process of a neural network, and in some cases improves the performance of the model via a modest regularization effect.

Keras provides support for batch normalization via the BatchNormalization layer.The layer will transform inputs so that they are standardized, meaning that they will have a mean of zero and a standard deviation of one.During training, the layer will keep track of statistics for each input variable and use them to standardize the data.

3.4.2 Bidirectional LSTM

Bidirectional LSTMs are an extension of traditional LSTMs

that can improve model performance on sequence classification problems. In problems where all timesteps of the input sequence are available, Bidirectional LSTMs train two instead of one LSTMs on the input sequence. The first on the input sequence as-is and the second on a reversed copy of the input sequence. This can provide additional context to the network and result in faster and even fuller learning on the problem.

Bidirectional LSTMs are supported in Keras via the Bidirectional layer wrapper.This wrapper takes a recurrent layer (e.g. the first LSTM layer) as an argument.It also allows you to specify the merge mode, that is how the forward and backward outputs should be combined before being passed on to the next layer.

In building this model, the following are defined : question text,question_stopwords and question_containdays are embedded using Embedding layer in Keras and batch normalization is used to train these sentences faster. LSTM have 3 layers operation inside it.

Dense layer implements the operation: output = activation(dot(input, kernel) + bias) where activation is the element-wise activation function passed as the activation argument, kernel is a weights matrix created by the layer, and bias is a bias vector created by the layer (only applicable if use_bias is True).

Flatten is used to flatten the input. For example, if flatten is applied to layer having input shape as (batch_size, 2,2), then the output shape of the layer will be (batch_size, 4). Flatten has one argument as follows keras.layers.Flatten(data_format = None)

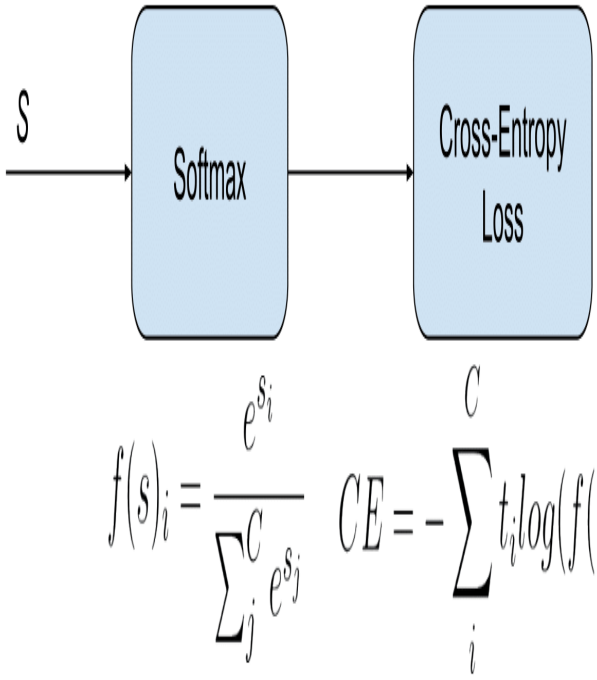
Activations can either be used through an Activation layer, or through the activation argument.

The rectified linear activation function or ReLU for short is a piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero.The range of this activation function is (0, inf), and it's not differentiable at zero.Its gradient is always equal to 1, this way maximum amount of the error can be passed though the network during back-propagation.

Softmax assigns decimal probabilities to each class in multi class problem. Softmax is implemented through a neural network layer just before the output layer. The Softmax layer must have the same number of nodes as the output layer.

If Dense(activation=softmax) is used then it will internally create a dense layer first and apply softmax on top it.It then shows us the result directly and the exact outputs of the last layer cannot be retrieved , instead, probability of occurrence is obtained.

Categorical Cross-Entropy Loss is also called Softmax Loss. It is a Softmax activation plus a Cross-Entropy loss.It is used for



for multi-class classification

Cross-entropy loss, or log loss, measures the performance of a classification model whose output is a probability value between 0 and 1. Cross-entropy loss increases as the predicted probability diverges from the actual label.

Fig 4 Categorical Cross Entropy loss

Adam is a replacement optimization algorithm for stochastic gradient descent for training deep learning models. Adam combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems.

4 RESULTS

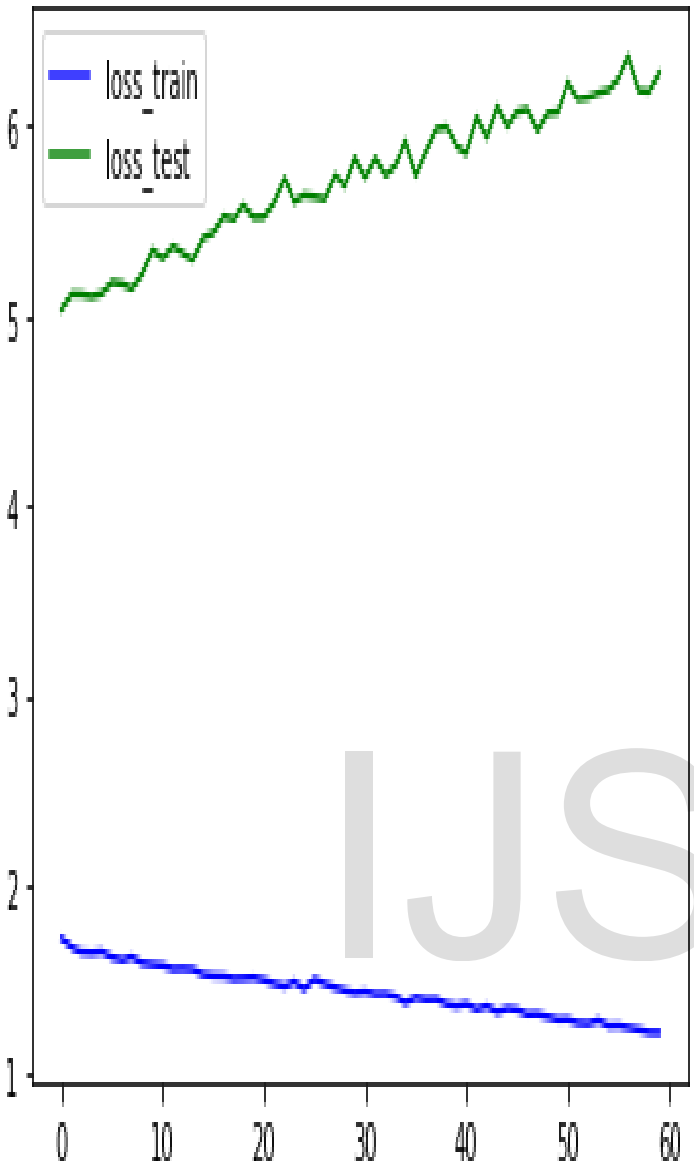
4.1 Evaluation Criteria

Data is divided into test data and train data.

The trained data is oversampled on less repeated question, which will help in increasing the dataset and not let it biased towards more repeated answers.

LSTM algorithm is applied on train data as well as test data.

As the model is trained using 3 LSTM layers , the accuracy obtained is more than the model that is trained using 2 LSTM lyers.



The following graph obtained shows loss on training and testing data.

Fig 5 Graph on Training and Testing data loss

The loss on testing data is more because testing data is not oversampled like the training data.

Accuracy :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Accuracy is the fraction of prediction. Accuracy is a metric for the performance analysis of the any prediction model. We can calculate accuracy by dividing the number of correct predictions with the total number of predictions . It determines the number of correct predictions over the total number of predictions made by the model.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Precision :

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

TP = True positive

TN = True negative

FP = False positive

FN = False negative

Recall :

Recall is the ratio of correctly predicted positive observations to the all observations in actual class.

F1 score :

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account.

F1 MICRO SCORE :

Micro F1-score (short for micro-averaged F1 score) is used to assess the quality of multi-label binary problems.

$$Micro-Precision=\frac{TruePositives1+TruePositives2}{TruePositives1+FalsePositives1+TruePositives2+FalsePositives2}$$

$$Micro-Recall=\frac{TruePositives1+TruePositives2}{TruePositives1+FalseNegatives1+TruePositives2+FalseNegatives2}$$

$$Micro-F-Score=2.\frac{Micro-Precision.Micro-Recall}{Micro-Precision+Micro-Recall}$$

It measures the F1-score of the aggregated contributions of all classes.

The classes obtained in this model are 155.

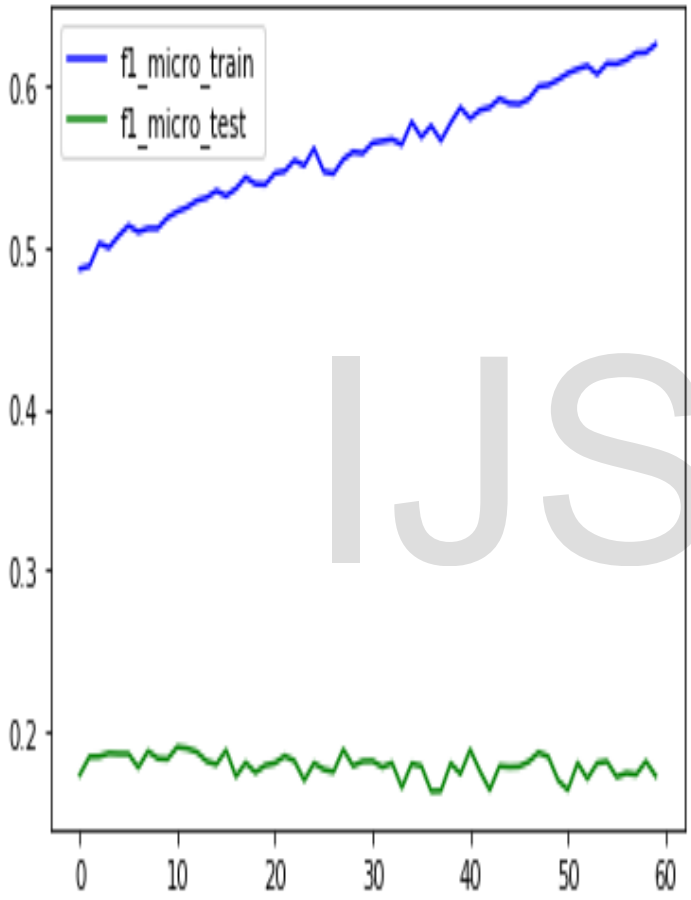


Fig 6 Evaluating training and testing data replies based on micro f1_score

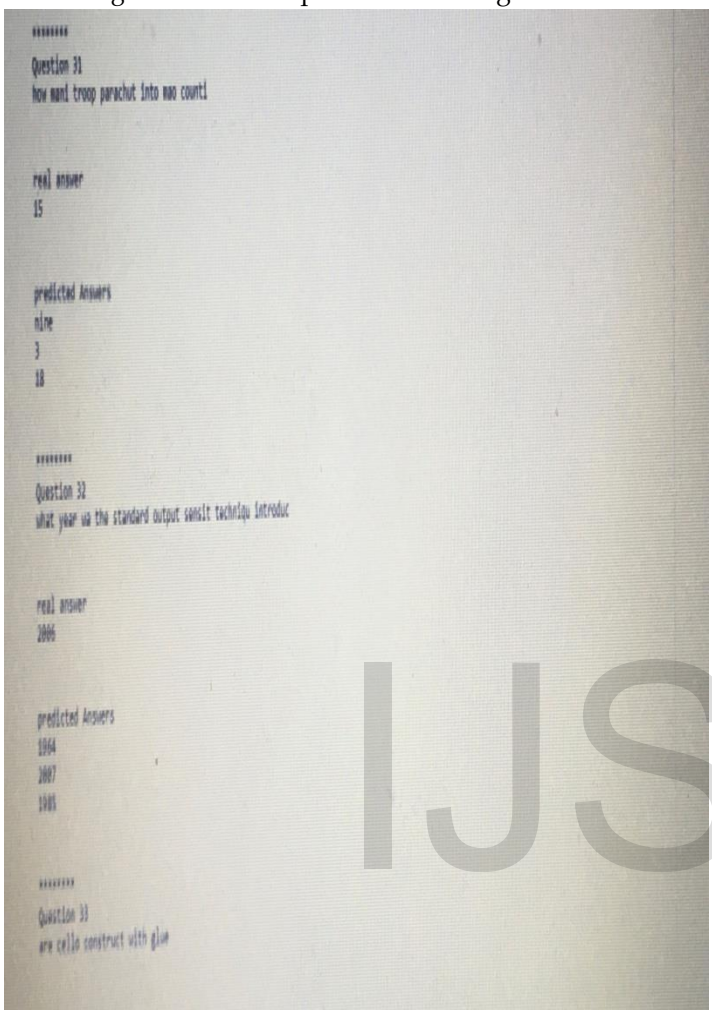
As training data is more oversampled than the testing data,the replies generated for these trained questions are predicted correctly.

4.2 **Output**

The following responses are obtained for the queries in the dataset.It mainly predicts the answer nearer to the actual reply.

TRAINING DATA – MORE ACCURATE REPLIES :

Fig 7 Predicted Replies For Training Data



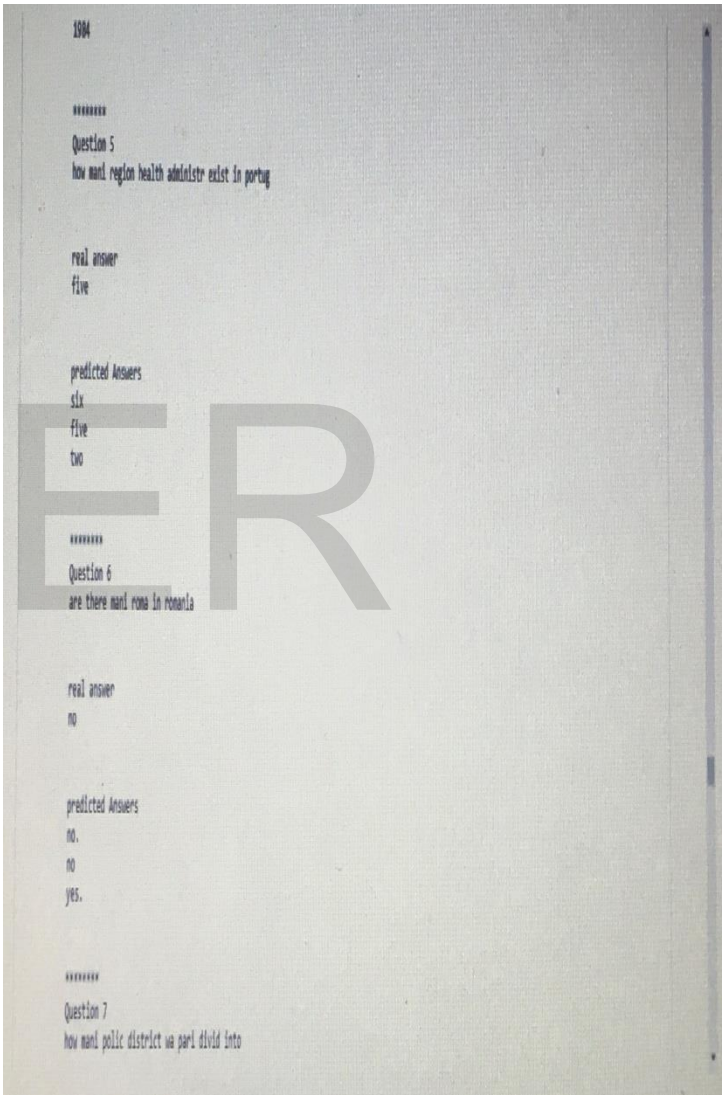
TESTING DATA – LESS ACCURATE REPLIES :

Fig 8 Predicted Replies for testing data

5 CONCLUSIONS AND FUTURE WORK

This proposed model gives an efficient prediction of responses which results in 60 percent accuracy and error of 40 percent approximately. It has been clearly demonstrated that LSTM has been successfully applied to predict responses. Even though the accuracy is less ,the replies generated are nearly of the same intention as model can't predict aptitute answers. So, the model tried to predict the best of the responses.

It is suggested to further improve the model reported in this study using more mail or chat content (e.g. non – information mails or personalized texts) to get a more realistic picture in predicting or forecasting responses. Responses are uncertain so we can not exactly predict future trends by using any model or theory but this model is very useful to take mitigation measures in advance by studying future trends to minimize



the incorrect response rate to certain extent and to implement it in Gmail and Chatbots.

REFERENCES

[1] Y. Artzi, P. Pantel, and M. Gamon. Predicting responses to microblog posts. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 602–606,

Montr´eal, Canada, June 2012. Association for Computational Linguistics.

[2] B. Pang and S. Ravi. Revisiting the predictability of language: Response completion in social media. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 1489–1499, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[3] L. Backstrom, J. Kleinberg, L. Lee, and C. Danescu-Niculescu-Mizil. Characterizing and curating conversation threads: Expansion, focus, volume, re-entry. In Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, WSDM '13, pages 13–22, 2013.

[4] A. Ritter, C. Cherry, and W. B. Dolan. Data-driven response generation in social media. In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, Edinburgh, UK, July 2011. Association for Computational Linguistics.

[5] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. Advances in Neural Information Processing Systems (NIPS), 2014.

[6] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural Computation, 9(8):1735– 1780, 1997.

[7] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[8] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals. Listen, attend, and spell. arXiv:1508.01211, abs/1508.01211, 2015.

[9] A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J.-Y. Nie, J. Gao, and B. Dolan. A neural network approach to context-sensitive generation of conversation responses. In In Proceedings of NAACL-HLT, 2015.

[10] L. Shang, Z. Lu, and H. Li. Neural responding machine for short-text conversation. In In Proceedings of ACL-IJCNLP, 2015.

[11] I. V. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau. Hierarchical neural network generative models for movie dialogues. In arXiv preprint arXiv:1507.04808, 2015.

[12] O. Vinyals and Q. V. Le. A neural conversation model. In ICML Deep Learning Workshop,2015.

[13] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15:1929–1958, 2014.

[14] J. Duchi, E. Hazad, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. JMLR, 12, 2011.