**SCIFI GENERATOR**

*Submitted in partial fulfillment of the requirements for the course of*

**INF 552-Machine Learning for Data Science**

*by*

## Sanjay Mallasamudram Santhanam

**3124715393**

**mallasam@usc.edu**

**Under the guidance of Prof.Satish Thittamaranahalli Ka**

### Viterbi School of Engineering,

### University of Southern California

April 2020

# Summary

The aim of this project is to build a text generator using different kinds of Recurrent networks like LSTM, bidirectional LSTM and stacked LSTM and compare their performances. The model is trained on a heterogenous mix of science fiction books from four different authors and also on one non-fiction science book. The models are trained on character sequences of text data and the output predicted is also a character.

|  |  |  |
| --- | --- | --- |
| **Serial number** | **CONTENTS** | **Page number** |
|  | Summary | ii |
|  | List of contents | iii |
|  | Figures list | iv |
| 1 | INTRODUCTION | 1 |
| 1.1 | Objective | 1 |
| 1.2 | Background | 1 |
| 1.3 | Long short term memory models | 1 |
| 2 | CHARAACTER LEVEL LANGUAGE MODELLING | 3 |
| 2.1 | Dataset | 4 |
| 2.2 | Model parameter | 4 |
| 2.3 | optimization | 5 |
| 3 | WORKING | 5 |
| 4 | COMAPRING RECURRENT NETWORKS | 7 |
| 4.1 | LSTM model | 7 |
| 4.2 | Bidirectional LSTM model | 7 |
| 4.3 | Stacked LSTM model | 7 |
| 5 | SAMPLE TEXT OUTPUT | 8 |
| 5.1 | LSTM | 8 |
| 5.2 | Bidirectional LSTM | 10 |
| 5.3 | Stacked LSTM | 13 |
| 6 | RESULT COMPARISONS | 15 |
| 7 | REFERENCES | 8 |

**Figures list**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Title** | **Page No.** |
| 1 | Model of LSTM cell | 2 |
| 2 | Model of Bidirectional LSTM cell | 3 |
| 3 | Accuracy of LSTM model | 9 |
| 4 | Loss of LSTM model | 10 |
| 5 | Mean Square error of LSTM model | 10 |
| 6 | Accuracy of BiLSTM model | 12 |
| 7 | Loss of BiLSTM model | 12 |
| 8 | Mean Square error of BiLSTM model | 13 |
| 9 | Accuracy of Stacked LSTM model | 14 |
| 10 | Loss of Stacked LSTM model | 15 |
| 11 | Mean Square error of Stacked LSTM model | 15 |
| 12 | Accuracy of Stacked all models | 16 |
| 13 | Loss of Stacked all models | 16 |
| 14 | Mean Square error of all LSTM models | 17 |

### 1. INTRODUCTION

**1.1 OBJECTIVE**

The main objective of the project is to study how text generation differs when using different kinds of LSTM models and compare their performances.

**1.2 BACKGROUND**

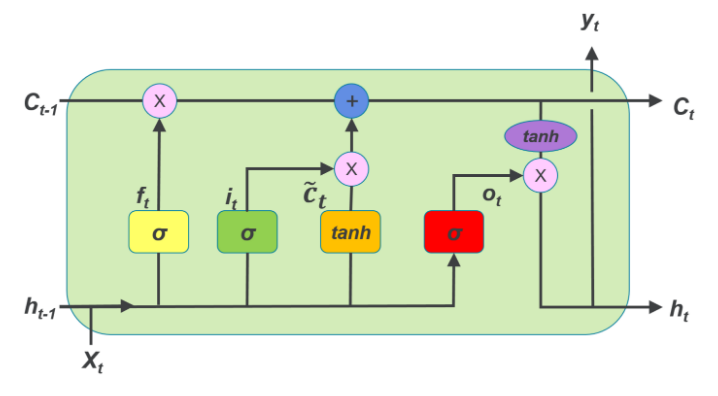
Recurrent Neural Networks(RNN) and Long Short-Term Memory (LSTM) have been widely used in problems that involve sequential data. Some applications RNN and LSTM are image captioning Vinyals et al. (2015); Karpathy & Fei-Fei (2015), speech recognition Graves et al. (2013), handwriting generation Graves (2013), neural machine translation Sutskever et al. (2014) and several sequential tasks Werbos (1988); Schmidhuber (2015); Rumelhart et al. (1985).

RNNs enjoyed early success but their applications were limited because of vanishing gradient descent problems. The practical difficulty of training even a small RNN models made were a challenge. LSTM Hochreiter & Schmidhuber (1997) were introduced to solve the exploding gradient problem.

An early work on LSTM is Hermans & Schrauwen (2013), which studies how Recurrent neural networks in character generation models learn long term interactions. It describes the success of recurrent networks in learning time series analysis.

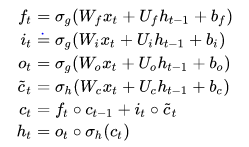
**1.3 LONG-SHORT TERM MEMORY MODELS**

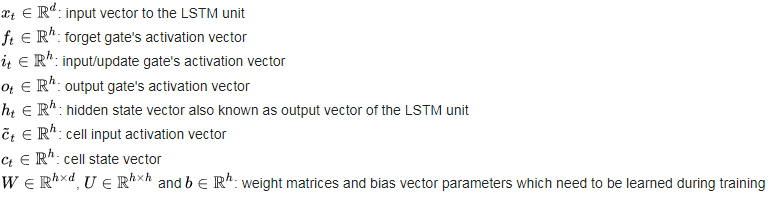
An LSTM cell consists of “cells” aka memory unit of LSTM and three gates namely input gate, forget gate and output gate that controls the data inside the LSTM neuron. The input gate determines how much of input is to be considered, the forget gate determines how much of input data to discard and the output gate determines how much of value in the cell can be outputted by the LSTM neuron. Usually, sigmoid is used as activation function of LSTM networks.



**Fig. 1** Model of LSTM cell

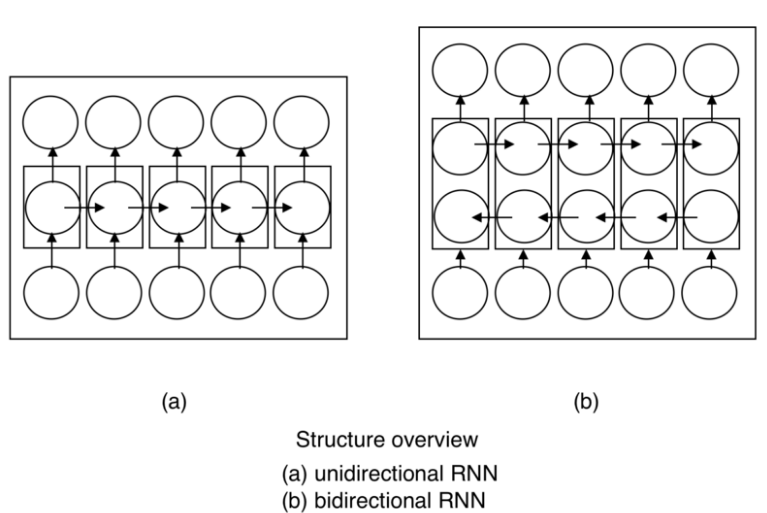
The mathematical description of LSTM cell is:





For Bidirectional LSTM, the cell is split into two directions, one for forward direction and other for backward direction. If all timestep inputs are available, Bidirectional trains one LSTM on input as-in and the other on a reverse copy of input. Those two states’ output are not connected to inputs of the opposite direction states.

Having such an architecture helps BiLSTM to model both the past and future information which is highly useful for text generation.



**Fig. 2** Model of Bidirectional LSTM cell

Stacked LSTM are nothing but multi layer LSTM with the addition that all the outputs at different timesteps of a cell are passed to the next layer, thus making the past outputs available.

**2. CHARACTER-LEVEL LANGUAGE MODELING**

There are two ways of generating text, word level text generation and character level generation. Although word level RNN models train faster, this project uses the character level text generation because they model the grammatical sequences correctly. Also character models have very smaller vocabulary and therefore require lesser memory during training and can learn patterns in shorter time.

In this project, 3 types of models using LSTM, Bidirectional LSTM and Stacked Bidirectional LSTM were trained on a set of science and science fiction books.

In all the models, the input is an embedded vector of character sequences and the output is one hot vector encoded softmax layer. If the timestep considered is T, and the vocabulary size is L, then the input is an encoded (T,L) dimensional matrix. Each character at each time step is encoded as a vector of size L.

The input data is trained to reduce the categorical entropy loss.

**2.1 DATASET**:

The books used as training data are obtained from Project Gutenberg (<https://www.gutenberg.org/>) , which has 60000 plus free downloadable e-books. The books used for this project are:

Science fiction books:

* All Around the Moon, by Jules Verne
* The Invisible Man, by H. G. Wells
* A Princess of Mars, by Edgar Rice Burroughs
* Worlds Within Worlds: The Story of Nuclear Energy, Volume 1 (of 3), by Isaac Asimov

Science book:

* The science of Interstellar by Kip Throne

The dataset is chosen to be from many authors because it learns many writing styles of different authors, as opposed to a single author. In order to get better realistic outputs, the model is trained on one science books so that it could learn some scientific terms which aren’t used much by science fiction authors

**2.2 MODEL PARAMETERS:**

The number of neurons in hidden layer is chosen to be 128/256 because they yield the lowest loss and also train quickly as opposed to a hidden layer size of 1024 neurons or 64 neurons. /

A rule of thumb is:

Nh=Ns /(α∗(Ni+No))

Ni = number of input neurons.  
No = number of output neurons.  
Ns = number of samples in training data set.  
α = an arbitrary scaling factor usually 2-10.

Here we have 50,000 samples approximately, Ni = 50(as we see 50 timesteps) and No= 46(vocabulary size). So if we have α = 2, we get Nh=50000/(2\*(50+46)) ≈ 260 . On the other hand if we have α=10,

Nh=50000/(10\*(50+46)) ≈ 50. To have a fine balance between the two extremes, a value of 128 seemed good.

Also the hidden layer size was experimented from 64 to 1024 (in increments of power of 2) neurons and surprisingly, Nh=128 gave best results.

For stacked LSTM, the 2nd hidden layer was experimented with from neurons 64 to 1024 neurons with an increment of a factor of 2 and a hidden layer of 256 neurons gave best results.

For the number of time stamps, although many of the previous implementations used 100 characters, the results had a lot of spelling mistakes. Using 50 characters reduced the number of word error, but the result had a bit of repeating words.

**2.3 OPTIMIZATION**

We use mini-batch stochastic gradient descent with batch size 128 and ADAM optimizer with base learning rate 1 × 10−3 and exponential decay for first moment and second moment to be 0.9 and 0.999 respectively. These settings work robustly with all of our models. The network is unrolled for 50 time steps and each model is trained for 30 epochs.

**3. WORKING:**

The model essentially learns character dependencies i.e. given a sequence of characters it learns the conditional probability distribution.

The dataset is converted to UTF-8 format to remove all extraneous UNICODE characters in the dataset like ¹,⁷,⁵,⁹,⁻,★,²,→,₁ which are totally useless for our current task. Only alphabets, numbers and few other characters like quotations, comma, fullstop etc. were considered.

When the dataset is loaded, Regular expression is used to filter characters and convert all alphabets to lowercase to reduce vocabulary size and also speed up the training.

Since neurons can only accept numerical inputs, the characters are encoded/mapped to a specific integer.

Then the dataset is prepared such that each character is converted to a one hot encoded vector i.e. if there as 50 timesteps and 46 characters, each input is of size 50\*46 and each character is converted to a sparse vector of size 46 with majority of the values being 0 with a 1 in the corresponoding index of the character. It is to be noted that one hot encoding of the input gives better accuracy than simple integer encoding. The input is finally converted to the form (number of training samples, timestamp, feature size).

Since the problem is a classification problem, the output is a predicted class of a character ie. Each character in considered as a class and the output label is also one hot encoded. This method is easier to train because it predicts the softmax probability of each character(soft classification), rather than predicting a single best character(hard classification). The standard categorical cross entropy loss is used as loss function.

When preparing the dataset, instead of having to take consecutive sequences, every third sequence is chosen so that the model doesn’t repeat learning almost the same input with different outputs.

Finally, the prediction is done based on diversity, which provides the output with more complementary values i.e. instead of just taking the argmax of the probability, we use a temperature value which decides randomness of the predictions by scaling down the values. Temperature factor is directly proportional to sensitivity towards lower probability classes i.e. it will make the model predict unlikely classes. Using a higher temperature generates a soft distribution thus making the model more diverse and also more prone to mistakes.

If temperature is high (τ→∞), then the probability of of all samples are almost the same and if the temperature is low (τ→0+), then the probability of the sample with highest value being selected is 1.

From the experimental results, a temperature around 0.3 seems to give a better output.

The models are trained on Google Cloud Platform with a single Tesla K80 GPU and 13GB memory.

**4.COMPARING RECURRENT NETWORKS**

**4.1 LSTM model:**

Model: "sequential\_4"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

lstm\_3 (LSTM) (None, 128) 89600

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_3 (Dropout) (None, 128) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batch\_normalization\_3 (Batch (None, 128) 512

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_3 (Dense) (None, 46) 5934

=================================================================

Total params: 96,046

Trainable params: 95,790

Non-trainable params: 256

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**4.2 Bidirectional LSTM model:**

Model: "sequential\_3"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

bidirectional\_2 (Bidirection (None, 256) 179200

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_2 (Dropout) (None, 256) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batch\_normalization\_2 (Batch (None, 256) 1024

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_2 (Dense) (None, 46) 11822

=================================================================

Total params: 192,046

Trainable params: 191,534

Non-trainable params: 512

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**4.3 Stacked LSTM model:**

Model: "sequential\_5"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

lstm\_4 (LSTM) (None, 50, 128) 90112

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_4 (Dropout) (None, 50, 128) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batch\_normalization\_4 (Batch (None, 50, 128) 512

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lstm\_5 (LSTM) (None, 256) 394240

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_5 (Dropout) (None, 256) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

batch\_normalization\_5 (Batch (None, 256) 1024

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_4 (Dense) (None, 47) 12079

=================================================================

Total params: 497,967

Trainable params: 497,199

Non-trainable params: 768

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**5. SAMPLE OUTPUT TEXT:**

Using the seed sentence, next 300 characters are predicted.

**5.1 LSTM:**

Epoch 00029:

Diversity= 0.3

Seed sentence= galaxy cluster abell 2218

gravitationally lenses m

e to the surface the same warped the particles to the same and should it was the moon and the contrations of the figure and the stars of the stranger than the moon, the control the contrance and the sight of the surface in the should the stranger of the stranger are the surface of the reasing with t

Diversity= 0.6

Seed sentence= galaxy cluster abell 2218

gravitationally lenses m

an astured into when i had great were and think in the contrations, and the particles and for the great see with

which a perfectly heresweressed and who you like the long to the handons had in the accurany of the companions the son of the pressones,

who was hall when he was being a project gutenberg

Epoch 00030:

Diversity= 0.3

Seed sentence= shared my sentiments in this respect i was positi

on of the consideration of the moon in for the moon is the stranger to the strange and the sun the streaks and the and to the search that she can the thing a that the stranger to the travellers to the atom and all the surface and the travellers and and the sun and the strange to the strange to the s

Diversity= 0.6

Seed sentence= shared my sentiments in this respect i was positi

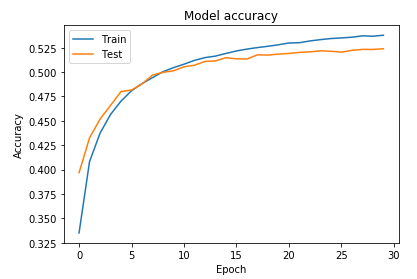
on and to see more spaces whotee, the body of the collest to the attraction of the second of a moring of the sunfices of the rapidly of the moon in the moon with the first and to the moon and she as a figure 28.2. and the bars. the table and i should be the hand enonging of the some thing

which he

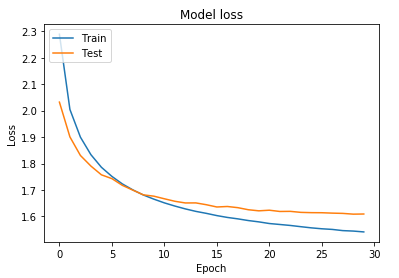
**Discussion:**

It is noted that lower the diversity value, lower is the number of incorrect spellings and higher is the repetition of articles(a,an,the),prepositions(of,to) and conjunctions(and) and frequently used words from training data(stranger is used many times in ‘Invisible man’, moon is used many times in ‘All around the moon’)

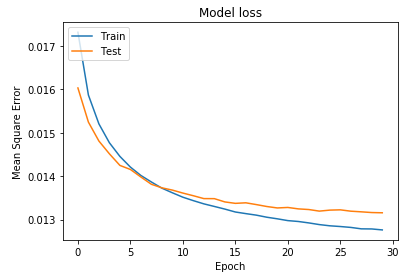
Some word sequence makes sense(‘the same warped the particles’ this follows after the seed sentence that contained ‘gravitational lensing’. It is a known fact that gravitational lensing of black hole warps the space around it and the model is also trying to predict the same) although some don’t(‘the stranger to the strange’ is meaningless and double repetition)



**Fig. 3** Accuracy of LSTM model



**Fig. 4** Loss of LSTM model



**Fig. 5** Mean Square Error of LSTM model

**5.2 Bidirectional LSTM:**

Epoch 00029:

Diversity= 0.3

Seed sentence= a million times that of the suns or

more, which me

the book of the stranger in the stranger of the endurance is the brand and the stranger for the possible was the tended around the lines and the content the sun of the rays of

the strength of the strong the formed the stranger was the consider the

moon of the stranger of the terrible and down the p

Diversity= 0.6

Seed sentence= a million times that of the suns or

more, which me

to come of over father and in a there from the shall gravitational

cane of the are world. the solar unters not found of his come of the most project gutenberg-tm rate and the or the epence was the string was the portion, when i had about a stars to present

moves the stranger in all the brigyte of h

Epoch 00030:

Diversity= 0.3

Seed sentence= in even a tolerable approximation of

the velocity

to the strange stor of the moon and the strength of the are and a reschene and the search and

all the officer that the projectile to the ordinary the lander of the surface of the south of the captain to the and

and explain the content of the moon as the moon in the strange higher that the centrifuga

Diversity= 0.6

Seed sentence= in even a tolerable approximation of

the velocity

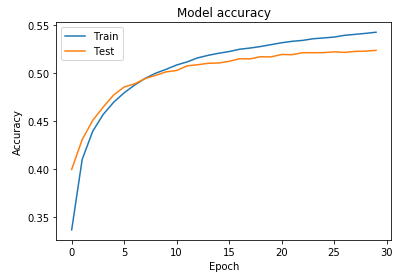
of the for a first we bolier the colver work,

the will be for the bear startations she the moon had the astracting rality for finally discuss the part of the red solar down you most struggle. the foundated had unother more in his computer and one

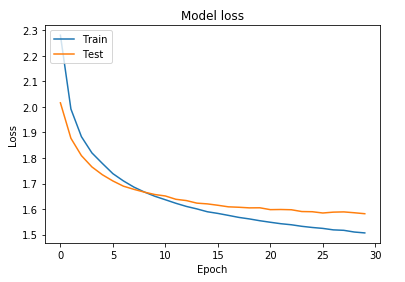
and bepoment as everything of the particles for the s

**Discussion:**

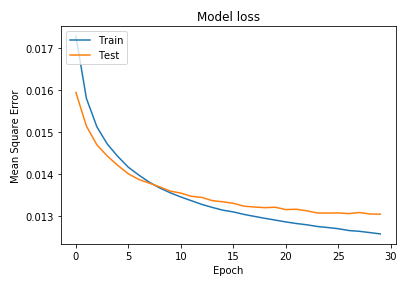
Bidirectional LSTM almost gives the same result as LSTM.. One reason could be the heterogeneity of the data. Since the training data is more diverse, it is difficult for BiLSTM to find the correlation.



**Fig. 6** Accuracy of BiLSTM model



**Fig. 7** Loss of BiLSTM model



**Fig. 8** Mean Square Error of BiLSTM model

**5.3 Stacked LSTM:**

Epoch 00029:

Diversity= 0.3

Seed sentence= ore frightened than hurt and more hungry than

eith

er travellers and the second beautiful atoms of the stranger so explained to the mass of the sun and the earth was a such a containing manner of the stranger control the first the projectile was a surface of the projectile and the distance of the stranger as the strength of the sun in the moon was a

Diversity= 0.6

Seed sentence= ore frightened than hurt and more hungry than

eith

er second. but a stranger, but a side

of the atoms were so the science warping of the planet that i sem, but the states would steel and and with this projectile as the and the anomalies and ropers possible and the sun to the country which is a mocness to first such a few particles, and then the new

Epoch 00030:

Diversity= 0.3

Seed sentence= observation.

"mountains chains are not numerous i

nterpretation of the strange and a second to the particles that the projectile and so for the sun, in a common and so many second black hole and for the constant singularity and the strange of the moon in the moon, and the services with a few more than the laws of the opposite than the bottom of the

Diversity= 0.6

Seed sentence= observation.

"mountains chains are not numerous i

nterstellar he had been made also starrication of

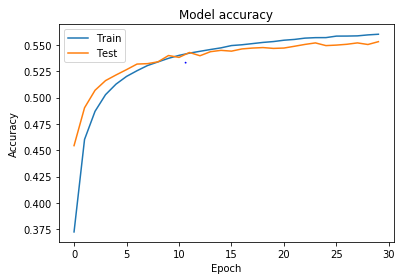
with a moon, it was now around a most resolution is a

start and moner is a community of the eyes and still be

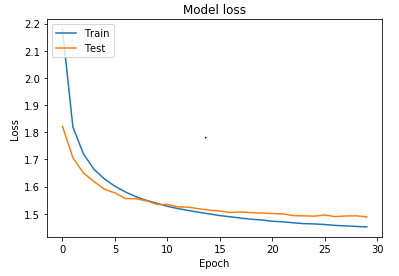
seed in a flat to explosion in preventing a common as the moon's and possible works of figure 29.6. the next and cities of the lunar consurt

**Discussion:**

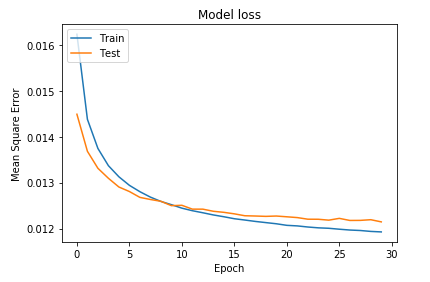
The stacked LSTM model yields longer meaningful sequences and lesser spelling mistakes than the other models. The usually repeating words are repeating less often and there is no double repetitions (‘the stranger to the strange’) like in previous models. It is a well-known fact that having more hidden layers improves accuracy but at the same time we shouldn’t add more than 3 or 4 hidden layers with LSTM cells as it might overfit and will lead to model remembering the text data instead of learning the conditional probabilities.



**Fig. 9** Accuracy of Stacked LSTM model

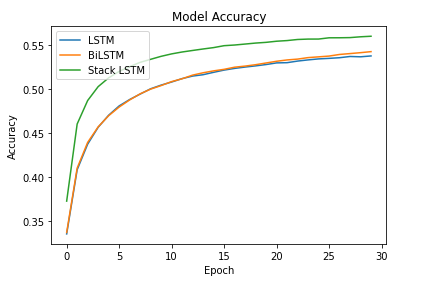


**Fig. 10** Loss of Stacked LSTM model

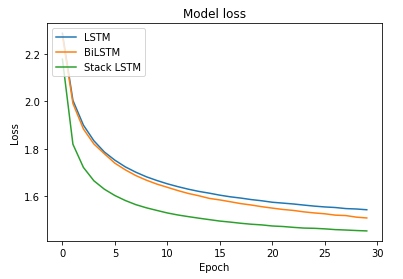


**Fig. 11** Mean Square Error of Stacked LSTM model

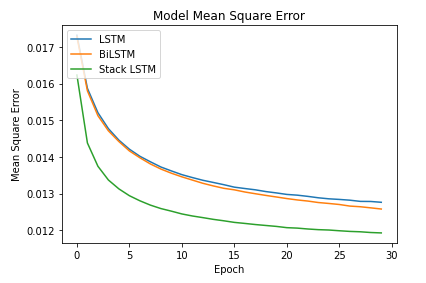
**6. RESULT COMPARISONS:**



**Fig. 12** Accuracy comparison of all models



**Fig. 13** Loss comparison of all models



**Fig. 14** Mean Square Error comparison of all models

|  |  |  |  |
| --- | --- | --- | --- |
|  | LSTM | BiLSTM | Stacked LSTM |
| epoch | accuracy | accuracy | accuracy |
| 0 | 0.335133 | 0.336725 | 0.372459084 |
| 3 | 0.456546 | 0.456997 | 0.502808273 |
| 6 | 0.488156 | 0.487671 | 0.525555849 |
| 9 | 0.504562 | 0.504041 | 0.537337184 |
| 12 | 0.514854 | 0.515947 | 0.544000268 |
| 15 | 0.521662 | 0.522533 | 0.549445987 |
| 18 | 0.526459 | 0.527798 | 0.552426934 |
| 21 | 0.530106 | 0.533219 | 0.555272579 |
| 24 | 0.534396 | 0.536738 | 0.556997061 |
| 27 | 0.537162 | 0.540465 | 0.558652163 |
| 29 | 0.537769 | 0.542672 | 0.560198665 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | LSTM | BiLSTM | Stacked LSTM |
| epoch | Loss | loss | loss |
| 0 | 2.288882 | 2.280267 | 2.179042 |
| 3 | 1.833562 | 1.819871 | 1.664069 |
| 6 | 1.722901 | 1.710315 | 1.580487 |
| 9 | 1.665372 | 1.65012 | 1.538621 |
| 12 | 1.628524 | 1.610433 | 1.51238 |
| 15 | 1.603234 | 1.58309 | 1.493572 |
| 18 | 1.584498 | 1.56152 | 1.480187 |
| 21 | 1.569642 | 1.542775 | 1.470608 |
| 24 | 1.556888 | 1.528028 | 1.462932 |
| 27 | 1.546453 | 1.516898 | 1.455668 |
| 29 | 1.541531 | 1.50675 | 1.451886 |

It is clear from the results that adding more hidden layers gives a better increase in accuracy than changing the cell type. BiLSTM gives better accuracy and lower loss than LSTM, but there is no significant difference. But Stacked LSTM yields better result than the other two models and there is visible difference in the predicted outputs.

**7. REFERENCES:**

Vinyals, Oriol, Toshev, Alexander, Bengio, Samy, and Erhan, Dumitru. Show and tell: A neural image caption generator. CVPR, 2015.

Karpathy, Andrej and Fei-Fei, Li. Deep visual-semantic alignments for generating image descriptions. CVPR, 2015.

Graves, Alex, Mohamed, A-R, and Hinton, Geoffrey. Speech recognition with deep recurrent neural networks. In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, pp. 6645–6649. IEEE, 2013.

Graves, Alex. Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850, 2013.

Sutskever, Ilya, Vinyals, Oriol, and Le, Quoc VV. Sequence to sequence learning with neural networks. In Advances in Neural Information Processing Systems, pp. 3104–3112, 2014.

Werbos, Paul J. Generalization of backpropagation with application to a recurrent gas market model. Neural Networks, 1(4):339–356, 1988.

Greff, Klaus, Srivastava, Rupesh Kumar, Koutn´ık, Jan, Steunebrink, Bas R., and Schmidhuber, Jurgen. LSTM: A search space odyssey. ¨ CoRR, abs/1503.04069, 2015.

Rumelhart, David E, Hinton, Geoffrey E, and Williams, Ronald J. Learning internal representations by error propagation. Technical report, DTIC Document, 1985.

Hochreiter, Sepp and Schmidhuber, Jurgen. Long short-term memory. ¨ Neural computation, 9(8): 1735–1780, 1997.

Hermans, Michiel and Schrauwen, Benjamin. Training and analysing deep recurrent neural networks. In Advances in Neural Information Processing Systems, pp. 190–198, 2013.