Creating Meaningful Short Story Using Deep Learning Neural Network

Kamesh Munuswamy
MSc in Data Analytics
x19199562@student.ncirl.ie
National College of Ireland

Prasanth Mariyagappan
MSc in Data Analytics
x19198744@student.ncirl.ie
National College of Ireland

Shivani Badola
MSc in Data Analytics
x19177127@student.ncirl.ie
National College of Ireland

Rajat Patil
MSc in Data Analytics
x19194048@student.ncirl.ie
National College of Ireland

***Abstract*—Story generation is a well-acclaimed activity in computational innovative analysis, but at the same time its challenging to evaluate empirically. It is always unreliable and expensive to rely entirely on human input to determine the content of stories produced. With Deep learning neural network being complex to study and unexplored due to their efficacy in performance, in this study we are highly motivated to generate meaningful short stories using Deep learning neural networks with an equal importance to model evaluation metrics. With an ideology to implement Machine learning techniques to handle the generation of textual data, in our research we have incorporated three following approaches such as LSTM, bi-LSTM Recurrent Neural Network model and GPT2 model for story creation by predicting the next word sequentially. We have made use of open source proof reading tool for comparative analysis and evaluation of model outcomes. Using this tool, we have validated the model performance by considering various evaluation metrics such as Accuracy, Grammaticality, Lexical Diversity etc., of all three approaches that is been implemented. The findings indicate some major variations between the models, were bi-LSTM model preformed convincingly well with an accuracy of 86 percentage indicating that this modelling approach may be useful for future studies.**

***Index Terms*—Short story, Deep learning, Neural networks, LSTM, bi-LSTM, Evaluation metrics**

I. Introduction

Creativity is one of the most fascinating area in computer science field of study. The most challenging and interesting question that is been raised in research of computing sector is that, can we design and program machines to be creative? In an effort to address this problem, a substantial amount of research is underway in the field of computer science innovation.

Using Machine learning and Deep learning techniques providing the strong platform to induce creativity in the field of computer science. In this research, we have implemented deep learning neural networks models to discover the capabilities of Neural networks. We have incorporated three approaches of deep learning recurrent neural network to create a meaningful short story without any human intervention. By training the hidden neural network with thousands of stories, the model is designed to predict the next subsequent word.

The work discussed here focuses on fictitious short story creation from the viewpoint of how human researchers construct stories by random word suggestions, but the aim is to concentrate exclusively on the development of imaginative outcomes. Story randomness without user interaction is a crucial concern for this research. This is the main purpose for using the 'random word' strategy to write a story [1].

In this study, we develop a neural network model which could produce a proper contextual grammar and a valid context phase. We can also use the neural network to create new stories which was not in the initial training dataset.

It's hard for the machine to write like a human. However, if the machine understands how to construct an essay, it may also compose an essay like a human. There are several researches that reflect words, grammar and semantics as a function. The Template Bag-of - Words (BOG) is a basic model used in natural language processing. Within this model, text is interpreted as a vector of words, that is, a word mean. Also, many analyses have been undertaken to produce text using vectors by learning to anticipate the next term in the recurrent neural network. Further exploratory study has shown that the Long-Short Term Memory Recurrent Neural Network (LSTM) generates better story than RNNs, as well as long-term dependence on appending a node to the window concept is learned to overcome gradient vanishing difficulties in RNNs [2].

One of the complications of story creation is how to quantify the content of stories empirically, because there is a vast amount of 'right' alternatives for even the subsequent sentence in a specified plot. Although measuring generation quality in a fully autonomous manner is likely to be as challenging as story generation, success in this work will benefit immensely from techniques that can offer a measure of generation quality without [11]. In this article, we had made use of open source proof reading tool to evaluate the quality of generated story of the model.

In aim of invoking the creativity using the computer, we have implemented three main deep learning neural network models to generate a short story. All the models used in this demonstration have been trained with thousands of stories with which neural network model learn the art of story formation and configuration. Using this technique, machine was able to predict the next corresponding word leading to create a meaningful short story. Each story generated by all three models have been validated and compared using important evaluation metrics.

II. Literature Review

[1] In this research, the author gives the explanation on Abstract Meaning Representation (AMR). AMR is nothing but the representation done in symbolic language where tens of thousands of English sentences are defined in. The paper provides an overview of the respective AMR and devices.

[2] In this research, the author gives the introduction on Penn Discourse Treebank's second edition (PDTB-2.0), wherein, he explains the annotations of lexical discourse relationships and gives the explanation of abstract object statements on huge word corpus. They take into consideration all elements of the analysis such as argument framework of the discourse relation, the sense annotation of that relationship, the attribution of the discourse relationship, and each of its arguments. They will provide descriptive statistics for the various facets of the annotation to the corpus.

[3] In this research, the author has given the demonstration of the event schemes introduced directly from the text corpus. There is some weakness in the analysis like several schemes do not have common topic and separate roles are mistakenly mixed into one another.

[4] In this research, the author has explored that it is hard to predict the word in the form of binary classification. This paper mainly aims on elimination of influence of the words that are predicted before. In this paper, they have utilized the CNN models where the percentage of accuracy is performed in character level while testing the efficiency. At the end, the result that we obtained by both machine learning and by CNN was good giving an accuracy of 80.10%.

[5] In this research, the author has suggested the new technique of next word prediction in language called Hindi. Here LSTM and Bi-LSTM were utilized for the next word prediction. Finally, the result that has been got at the end was the accuracy of 81.07% for BI-LSTM and 59.46% for LSTM. This method can be utilised as an application for automatically completion of sentence, automatic completion of stories, etc.

The author had mentioned a few techniques in this paper [6], including TAIL-SPIN, MINSTREL, and BRUTUS, which were developed about 15 years ago to produce the random terms. The author had also mentioned that his research was using the "Expert Systems" approach. Expert system is a standard computer program that simulates how humans think about solving complex issues. In addition, WordNet, ConceptNet libraries are often used to test the grammar and create a sentence that may make sense. With this method the author succeeded in developing a model that could construct a story by sequentially predicting the words.

In a further study [7], the author contrasted two simple models, namely the Recurrent Neural Network Encoder Decoder (RNNED) and the Recurrent Neural Network for Story Generator (RNNSG). In this study, the model performed much better than the latter, since RNNSG was constantly aware of the sentences created by the preceding stage and was also able of build more sentences with a source meaning. Performance was measured using the ROUGE-N test and it reported that RNNSG had greater precision and ROUGE-N performance than RNNED.

Few more studies [8] note that the Recurrent Neural Networks are the state-of-the-art paradigm for fulfilling the task of generating stories. The author, however, had demonstrated that Multiplicative Recurrent Neural Network (MRNN) performs much better than standard RNN models. The author states that hidden-to-hidden matrix and inputs depend upon the dynamics of RNN's hidden states. The model would initially take input values through the input layer and add weight in the visible-to-hidden layers. Factorizing the inputs on this layer is called MRNN, which works best for longer memorizing the meaning and predicting the next term for the same meaning.

This paper [9] explores a new method for testing the comprehension of the story, the generation of stories, and the learning of scripts, called the 'Story Cloze test'. This method requires a program to determine the proper ending to a tale of four sentences. Few other models, such as Skip-thought Model for embedding Sentence2Vec and Deep Structured Semantic Model (DSSM). DSSM did, however, provides greater accuracy than other models. Compared to previously used Narrative Close Test which performed worse than randomly guessing the Story Close Test acted as an important assessment for learners of both the comprehension of plot and the knowledge of script.

The author outlines a series of strategies for doing the Story Cloze Test in the current paper [10]. The highest test result is 67.2 percent accuracy, outperforming the highest baseline of 58.5 per cent. Initially, the recording of two specific unsupervised baselines used in some story-predicting exercises. Instead describe the supervised approach, which uses an artificially generated recurrent neural network (RNN) with a binary classifier to distinguish appropriate story endings from incorrect endings. Comparing outcomes of this model when alternately conditioned on specific story encoding and various techniques to achieve incorrect endings.

[11] The author presents Sentence-BERT(SBERT) in this publication, a modification of the pre-trained BERT network that uses siamese and triplet network architectures to extract semantically accurate sentence embedding that can be calculated using cosine-likeness. This reduces the effort of identifying the most similar pair while retaining BERT 's accuracy from 65 hours with BERT / ROBERTa to around 5 seconds with SBERT. We check SBERT and SRoBERTa on different STS tasks and transformation learning tasks that are used to perform certain state-of-the-art sentence embedding methods.

In this research [12], the RNN technique on the basis text summarization has been utilised. The method is known as story scrambler. Based on the stories that are already fed to the model in the beginning, the model focused to create the new stories. To create the new stories, they have considered two factors. They are firstly, they have chosen only those stories which consists of different story line and different characters. Secondly, they have chosen those stories which have similar story line and similar characters. Evaluation of these stories that are generated from this model is verified from grammar checker.

In this research [13], the author shows how Bidirectional Encoder Representations from Transformers (BERT) can be utilized in generation of stories. On the basis of BERT, they have developed the new document-level encoder that can express a document's semantics and obtain exHere, they introduce the techniques of query-oriented single-document summarization by utilizing deep auto encoder (AE). Here the analysis of vocabularies which are local as well as global is performed. Here the random noise is added to the local term frequency and the effect of this is experimented as the input expression of the auto encoder. By utilizing the auto encoders of vocabularies that are local, the results demonstrate discriminatory feature space. They also show that the performance of the summaries generated can be further enhanced by a two-stage fine tuning method.

In this article [14], they introduce the techniques of query-oriented single-document summarization by utilizing deep auto encoder (AE). Here the analysis of vocabularies which are local as well as global is performed. Here the random noise is added to the local term frequency and the effect of this is experimented as the input expression of the auto encoder. By utilizing the auto encoders of vocabularies that are local, the results demonstrate discriminatory feature space.

In this paper [15], survey has been done on to explore the methods that perform single and multiple document summarization. The special importance has been given to the extracting techniques. Discussion on certain important methods that deals with the specific information of the text summarization has been performed. Importance is given to the automatic testing of the generated tests.

In this paper [16], the author demonstrate about the method of unique text generation for mobile devices and its evaluation methodology. This method is context driven from location, weather, scheduled events, etc.). Implementation of this approach also finds its limitations and constrains by means of computational resources and usage of data but due to the subjective creative task activities produced multiple possible outputs. In this approach, coherence metrics was introduced to deliver greater quality by means of human perspective. This subjective approach was able to attain readability accuracy of 86% on comparing the spearman correlation metrics to human assessment text.

In this research [17] of another interesting automatic text generation using deep learning, the author has proposed attention mechanism to narrate a story using machine learning. On implementing the attention mechanism there are two models implemented in this study such as Syntax-Guided Machine Reading Comprehension (SG-Net) to learn and understand the Chinese word vectors semantics and semi-supervised self-growing generative adversarial network (SG-GAN) to predict and generate realistic sequences of Chinese words. Specific set of evaluation experiment which uses BLEU has been performed to validate the model outcome. Its been has studied that SG-GAN performed as expectation than SG-Net on generating the next sequence of words.

In another research paper [18] on the text prediction, the author proposes an unique approach of using multi-label attributes to structure the text and classifying the problem based on statistical data. Using with technique the relationship between the text are identified by using entropy method which makes a stronger association of text. By the collaborating the functionality of multi-label attribute of text and entropy method, the proposed model predicted the text with a guaranteed better accuracy.

From this paper [19] which uses integrated narrative generation system for story generation which create a story by forming a discourse tree structure with diverse story techniques. In this approach both micro level and macro level story level techniques are collaborated flexibly to create a different types of story semantics. From this study, through combination of two different techniques it is been found that it is equal to give equal importance both the techniques individually as well as when they are fused flexibly.

This paper [20] discusses the FASTY language feature, a text forecasting program developed to increase the quality of text communication for people with disabilities. The FASTY language aspect is focused on state-of-the-art n-gram-based word-level and part-of-talk projections and on a variety of creative modules designed to improve output in various languages.Along with its system structure, these new technologies make it flexible to a multitude of languages without affecting the quality and performance. The framework can be quantified for various interfaces and with a number of different applications.

III. Methodology

In the process of implementation, the initial and a crucial stage is interpretation of data to understand the data patterns. In order to accomplish the knowledge discovery of data used, we have implemented one of most followed methodology in data mining called Knowledge Discovery in Databases (KDD).

A. KDD Methodology

KDD methodology is a process of finding and interpreting data patterns by performing repeated application of iterative steps.

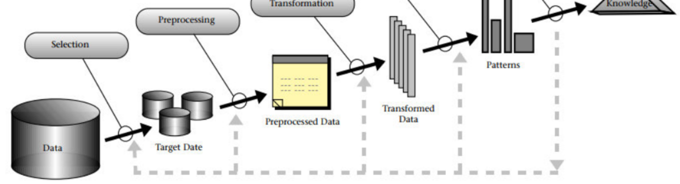

Fig. 1. KDD Methodology

- **Data cleaning:** Data cleaning is described as a reduction of irrelevant and redundant data from the collection of data (Dataset/metadata).
- **Data integration:** Data integration is represented as heterogeneous data across multiple sources merged into a centralized source (DataWarehouse).
- **Data selection:** The data selection is characterized as a process in which the data related to the study is determined and extracted from the data set.
  - Data selection using Neural network.
  - Data selection using Decision Trees.
  - Data selection using Naive bayes.
  - Data selection using Clustering, Regression etc.
- **Data Transformation:** Data Transformation is known as the process of transforming data into the correct form demanded by the mining method.
  Data Transformation is a two step process:
  - Data Mapping: Assigning components from source to destination to record transformations.
  - Code generation: Creation of the actual program for transformation.
- **Data Mining:** Data mining is classified as practical techniques that are used to find potentially valuable patterns.
  - Transforms the related activity data into trends.
  - Decides the purpose of the model through classification or characterization methods.
- **Pattern Evaluation:** Pattern Evaluation is represented as recognition of explicitly increasing trends of information dependent on the measure.
- **Knowledge representation:** Knowledge representation is described as a method that incorporates visualization tools to depict the effects of data mining results.

B. Data Source

This dataset is collected from the "Big Bad NLP Archive" established by Quantum Stat.com, that provides enterprise applications for natural language processing (NLP) and data science.There are approximately 100,000 + records in the dataset, which include five sentences based on various stories and a story title.
Project Dataset Link: [https://cs.rochester.edu/nlp/rocstories/].

C. Data Preprocessing

The dataset contains five sentences and a story title taken from various stories. All sentences are merged into a column called sentences. From which 5000 records are taken. For pre-processing, the Keras deep learning library is being used which contains a Tokenizer class which is used to encode the text data to unique numbers. The tokeniser goes through all the sentences and uses the fit_to_text method to fit on texts to generate numbers. It supports a method called text_to_sequences that generates a sequence of tokens represents every sentences.

Each text document in the dataset is converted into a sequence of tokens representing the ngram phrases generated from the input data, where each integer corresponds to the index of a specific word in the complete word vocabulary of the source text.

Now, it is possible that different sequences have different length. For handling this case, padding should be done on the sequence to make their length equal. Pad_sequence function in keras is used to accomplish the mentioned task. To input this data into the learning model, predictors and labels are created. Creating N-gram sequence as predictors and the next word of the N-gram as label.

IV. Exploratory Data Analysis (EDA)

In this section we have performed some high level EDA on our given data. By this analysis we have inferred crucial data insights which will be greatly help in model structuring.

A. Story genre EDA

The below mentioned graph is plotted using different types of story genre available in the dataset. This graph displays the top 20 stories and their count available in our dataset. The dataset contains various story genres with equal distribution in count as well. This analysis give us an idea about our dataset story genre which helps us to feed the model accordingly.

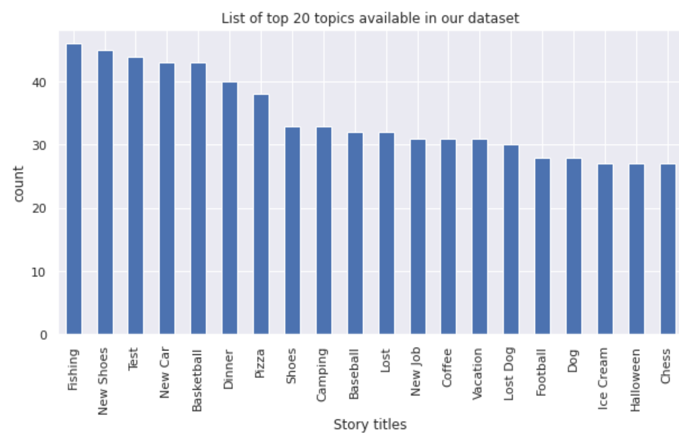

Fig. 2. EDA- Story genre

B. Word could EDA

We have performed word cloud EDA to understand the common and different types of words available in our dataset. The dataset consist of millions of texts as a story. The data is composed of short stories with five separate sentences. The model is trained with this thousands of different genre stories which helps the model to learn and understand the story sematics and flow.

We have designed the model to predict a next sequenial word by initially passing a seed word. Word cloud simply plots the different and most common words available in the dataset. This analysis helps to picturise the seed word which has to be passed to as an input based on which the model predicts the next sequential words to create a story.

Fig. 3. EDA- Word Cloud

C. Sentence length EDA

This EDA is mainly carried out to identify the input structure based on which padding of the input is structured. In tensor flow it is important to flow the same input sequence. From the analysis its been found that the maximum sentence length is 69. The padding is carried out to create an input array of 69 characters length for each input which is then feed to the model.

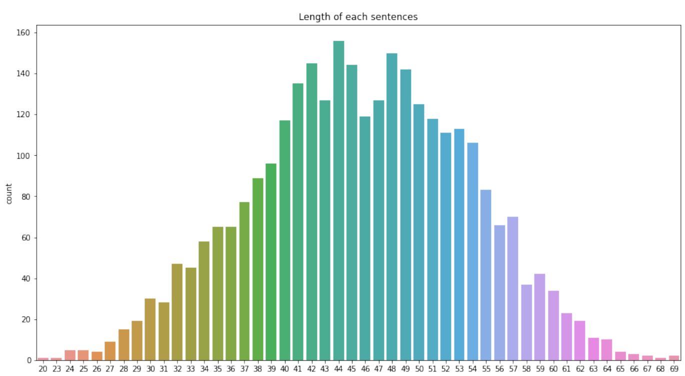

Fig. 4. EDA- Maximum Sentence length

V. Neural Network Models and Implementation

In this project we have followed three approaches leading to implement three different neural network models.

- **Approach 1: Recurrent Neural Networks-Long Short-Term Memory (RNN- LSTM)**
  Recurrent Neural Networks-Long Short-Term Memory (RNN- LSTM) will be used to create a new text sequence as it generates sentences based on a given input text. Recurrent Neural Networks dominates complex problems of machine learning including text processing and generation. Such an architecture enables the network to learn and generalize through input sequences, rather than defining individual patterns. We used RNNs not as predictive models but as generative models. This means our network will learn the sequence of a text and then produce a completely new possible word which matches the entire context.
- **Approach 2: GPT2 Model**
  GPT2 is a huge, Web Text-trained Transformer language model, a diversified corpus available in internet text. It encompasses more than 80 million documents which is equivalent to the overall text data of 40 GB. The model contains 1542 million variables which gives the outstanding results on various languages. The model that we have used in our project is the full-size GPT2 model. It contains 12 different layers and 117 million variables. It contains the vocabulary which has of 50,257 byte-pair-encoding tokens. This encoding permits the model to parse and create new Unicode string irrespective vocabulary size. The model can handle the data which is in text which is of length of approximately of 1024 BPE tokens.
  There are different types of decoding methods. They are as follows.
  Greedy search: The word which has the highest probability will be selected in the greedy search. Here in our project, we have passed the text which is present in the data set ('Tom had a very short temper.'). The words that are created following the text that has been passed is reasonable. But the drawback is that model fast begins to repeat itself. This drawback has been overcome using the Beam search.
  Beam search: This method lowers the risk of losing the unseen high probability sequences of words by retaining the most probable num beams of hypothesis at every single time step and ultimately selecting the one which has the maximum probability. Beam search usually produces the sequence of output with the probability better than the greedy search method. But there is no guarantee that the output will be the one which we have desired. The advantage of the Beam approach is that there can be comparison of the top beams and can select the beam which meets our requirement.
- **Approach 3: Bidirectional LSTM (BiLSTM)**
  In this approach, a recurrent neural network architecture is being implemented which is called a bidirectional LSTM (BiLSTM). This model outperforms well with processing text data where a context needs to be maintained throughout a sentence. Additionally, an embedding layer is also used as a first layer which helps to get an n-dimensional vector for each word embeddings. This would help to enhance the efficacy of BiLSTM layers to identify the similar words of a sentence and maintain the context throughout the sentences.

VI. Evaluation and Result

All three models implemented in this study is trained with dataset comprising of thousand different stories in sentences. All the five sentences are merged together to form a story. This merged sentences are being used to train the models from which the models learn and understand the story configuration and structure.

	storyid	storytitle	sentence1	sentence2	sentence3	sentence4	sentence5	sentences
0	8bbe6d11-1e2e-413c-bf81-eaea05f4f1bd	David Drops the Weight	David noticed he had put on a lot of weight re...	He examined his habits to try and figure out t...	He realized he'd been eating too much fast foo...	He stopped going to burger places and started ...	After a few weeks, he started to feel much bet...	David noticed he had put on a lot of weight re...
1	0beabab2-fb49-460e-a6e6-f35a202e3348	Frustration	Tom had a very short temper.	One day a guest made him very angry.	He punched a hole in the wall of his house.	Tom's guest became afraid and left quickly.	Tom sat on his couch filled with regret about ...	Tom had a very short temper. One day a guest m...
2	87da1a22-df0b-410c-b186-439700b70ba6	Marcus Buys Khakis	Marcus needed clothing for a business casual e...	All of his clothes were either too formal or t...	He decided to buy a pair of khakis.	The pair he bought fit him perfectly.	Marcus was happy to have the right clothes for...	Marcus needed clothing for a business casual e...
3	2d16bcd6-692a-4fc0-8e7c-4a6f81d9efa9	Different Opinions	Bobby thought Bill should buy a trailer and ha...	Bill thought a truck would be better for what ...	Bobby pointed out two vehicles were much more ...	Bill was set in his ways with conventional thi...	He ended up buying the truck he wanted despite...	Bobby thought Bill should buy a trailer and ha...
4	c71bb23b-7731-4233-8298-76ba6886cee1	Overcoming shortcomings	John was a pastor with a very bad memory.	He tried to memorize his sermons many days in ...	He decided to learn to sing to overcome his ha...	He then made all his sermons into music and sa...	His congregation was delighted and so was he.	John was a pastor with a very bad memory. He t...

Fig. 5. Dataset with Merged Sentences

- **Approach 1: Recurrent Neural Networks-Long Short-Term Memory (RNN- LSTM)**
  This model is trained with the two datasets, x_value which contains words, from the input file, of a particular sequence length; y_values which contains the next word corresponding to the sequence in x_value. The data in x_value is stored by mapping the individual words of the sequence with their respective indices from the dictionary generated in input processing module. x_value is then reshaped and normalized. The second dataset, y_values have categorical features so, one hot encoding is used to transform the categorical features into a format that would better work with machine learning algorithm. In this, we create an array where the element at that index which corresponds to the respective word in the dictionary is set as 1 while the elements at the remaining indices are kept as 0.
  Now, defining the LSTM model. It will contain 400 memory units (neurons). This model is trained around 44623 sentences for 30 epochs which has given **accuracy of 31%** to create a meaningful story. The story generated by this model is not meaningful. Adding more data or fine tuning the network parameters accuracy of the model can be improved further. However, we will be using Bidirectional LSTM for better accuracy.
  The story generated by LSTM model is evaluated using open source Grammarly tool which validates the short story using general metrics. The tool evaluates the story for grammar, spell check, punctuation check, word count etc. On validating the story with general metrics, LSTM model generated story obtained an overall **score of 71%**.

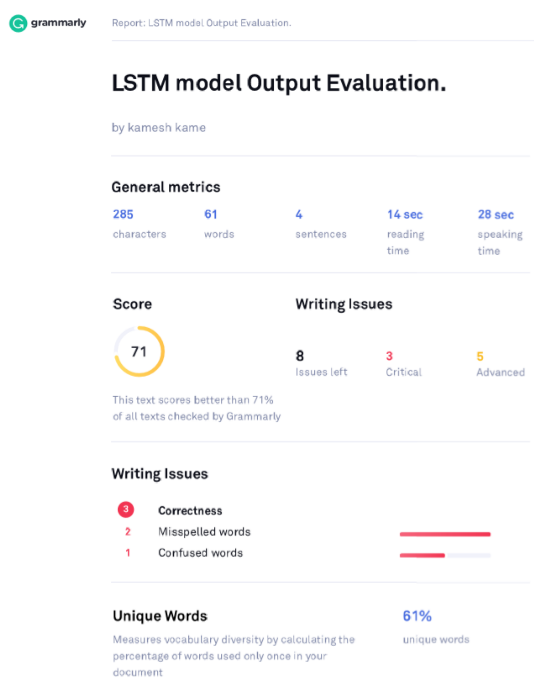

Fig. 6. Grammarly Result Evaluation Report 1- LSTM Model

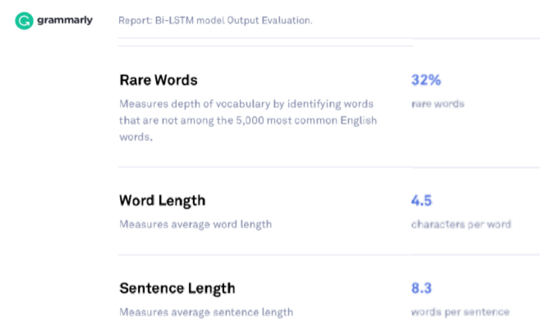

Fig. 7. Grammarly Result Evaluation Report 2- LSTM Model

- **Approach 2: GPT2 Model**
  On evaluating the story generated by Web Text-trained Transformer language GPT2 model with Grammarly tool, the model output has an **overall score of 42%**. The main reason behind this below average score is capitalization of words. Other than that the story generated by the

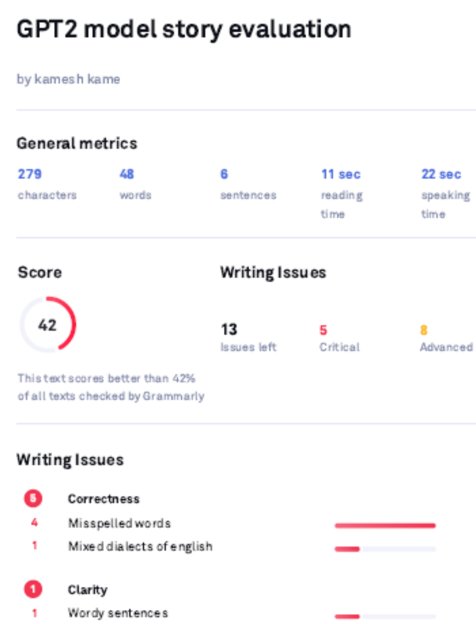

Fig. 8. Grammarly Result Evaluation Report 1- GPT2 Model

model was quite satisfying with proper grammar and punctuation's.

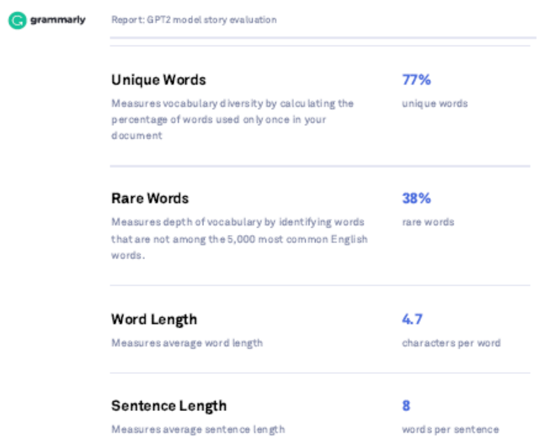

Fig. 9. Grammarly Result Evaluation Report 1- GPT2 Model

- **Approach 3: Bidirectional LSTM (BiLSTM)**
This network is trained with around 10000 sentences for 52 epochs which gives an **accuracy of 86%** to create a meaningful story. During training, Adam optimizer plays a vital role to alter the weights and bias of each neurons of the BiLSTM layers in order to find the local minimum where the loss would be less. While fitting the model, embedding layer would also be trained on each epoch which improves the performance of the model drastically. An opensource python package called TensorFlow is being used throughout this process starting from building an architecture of a neural network to train and deploy it.
Since TensorFlow expects the input shape should be equal throughout the training and prediction process, this model is limited to generate up to 69 consequent words. This is because, all sentences were having in and around 69 words. Hence an array of size 69 had been taken as constant and preprocessed the data using several preprocessing techniques including padding to match with the input shape.
During prediction, a seed word needs to be given as input by padding the remaining words of an input array as zero. Embedding layer would convert the word to an n-dimensional vectors and the BiLSTM would predict a word. Then this word would be appended with the seed word and then fed into the model again post padding. Iteratively this would keep predicting up to 60 words which would be concatenated together to create a meaningful story with proper context.
The evaluation report of grammarly open source tool for the story created by Bi-LSTM model produced an **overall score of 98%**.The story generated by Bi-LSTM model was more readable and meaningful which out-powered the results of other models.

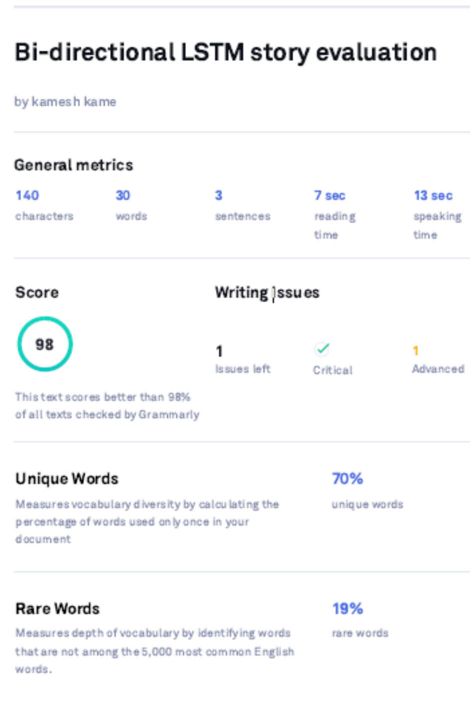

Fig. 10. Grammarly Result Evaluation Report 1- Bi-LSTM Model

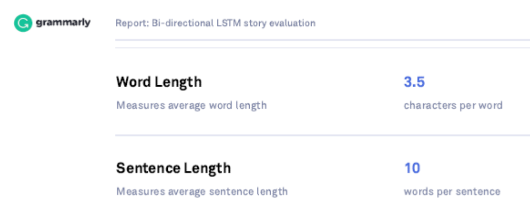

Fig. 11. Grammarly Result Evaluation Report 2- Bi-LSTM Model

The blow chart is plotted against accuracy and losses with respective to epoch count. It is been observed that the losses reduced rapidly in increase in the epoch count. Mean while the accuracy was constant after a certain epoch count run.

VII. Conclusion and Future Work

With the aim of invoking creativity using machine programming in the field of computer innovation research. We have made use of the advancement of data mining and deep learning neural network platform to create a meaning short story by training the neural network models by thousands of different story. To attain our goal in creating the meaningful stories we have approached and implemented three different models such has Recurrent Neural Networks-Long Short-Term Memory (RNN- LSTM), GPT2 model and Bidirectional- LSTM model. We have Incorporated open source Grammarly tool has a part of evaluation metrics. As a part of evaluation of the stories generated by all three models, Bi-LSTM model produced

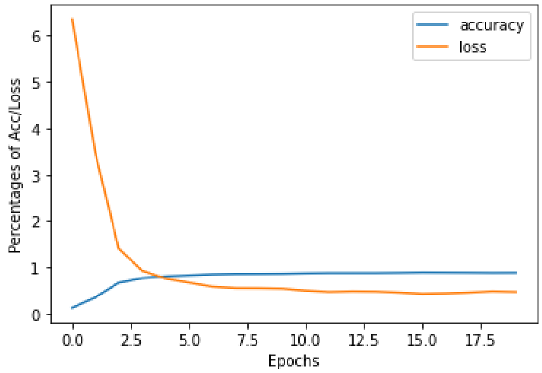

Fig. 12. Accuary and loss curves- Bi-LSTM Model

an outstanding result of 86% accuracy highest among all. Whereas, LSTM model was managed to produce an accuracy of 71%. Using Grammarly tool as a part of evaluation which was handy in validating the grammatical metrics and lexical structure.

It will be fascinatingly interesting to enhance the model behavior to create much better meaningful stories. Which can lead us to write or create full length creative stories using neural network models as a part of future work. This can done by training the models with millions of stories of all genre.

REFERENCES

[1] Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider, "Abstract meaning representation forsembanking," London, vol. A247, pp. 529–551,2013.

[2] Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K. Joshi, and Bonnie L. Webber, The pen discourse treebank. In LREC. European Language Resources Association, vol. 2. 2008.

[3] Niranjan Balasubramanian, Stephen Soderland, Oren Etzioni Mausam, and Oren Etzioni, "Generating coherent event schemas at scale," In EMNLP, 2013, pp. 1721–1731.

[4] A. Basu, A. Garain and S. K. Naskar, "Word Difficulty Prediction Using Convolutional Neural Networks," IEEE Region 10 Conference (TENCON), Kochi, India, 2019, pp. 1109-1112,doi: 10.1109/TENCON.2019.8929231.

[5] R. Sharma, N. Goel, N. Aggarwal, P. Kaur and C. Prakash,, "Next Word Prediction in Hindi Using Deep Learning Techniques," International Conference on Data Science and Engineering (ICDSE), Patna, India, 2019,pp. 55-60, doi: 10.1109/ICDSE47409.2019.8971796.

[6] Richard S. Colon, Prabir K. Patra, Khaled M.Elleithy, "Random word retrieval for automatic story generation," Proceedings of the Zone 1 Conference of the American Society for Engineering Education, vol. 2, pp. 740–741, 2014.

[7] YunSeok Choi, Suah Kim, Jee-Hyong Lee, Recurrent Neural Network for Storytellin, 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS), 2016.

[8] Ilya Sutskever, James, Geoffrey Hinton, Generating Text with Recurrent Neural Networks, University of Toronto, 6 King's College Rd, Toronto, ON M5S 3G4 CANADA. 2016.

[9] YunSeok Choi, Suah Kim, Jee-Hyong Lee, Recurrent Neural Network for Storytellin, 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS), 2016.

[10] YunSeok Choi, Suah Kim, Jee-Hyong Lee, Recurrent Neural Network for Storytellin, 2016 Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS), 2016.

[11] Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli and James Allen, Human Language Technologies (NAACL HLT), 2016 Conference of the North American Chapter of the Association for Computational Linguistics, 2016.

[12] Yousefi-Azar, M. and Hamey, L. Text summarization using unsupervised deep learning, Expert Systems with Applications, 68, 2017 , pp. 93–105.

[13] Liu, Y, Lapata, M.Text summarization with pretrained encoders,EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, 2019 , pp. 3730–3740. doi:10.18653/v1/d19-1387.

[14] Pawade, D, Sakhapara, A, Jain, M, Jain, N and Gada, K, Story Scrambler - Automatic Text Generation Using Word Level RNN-LSTM, International Journal of Information Technology and Computer Science, 10(6), 2018, pp. 44–53.

[15] Nazari, N., Mahdavi, M.A survey on Automatic Text Summarization. Journal of AI and Data Mining, Expert Systems with Applications, 68, 2018 , pp. 43–60, doi:10.22044/jadm.2018.6139.1726.

[16] Marchenko, O., Radyvonenko, O., Ignatova, T., Titarchuk, P. and Zhelezniakov, D., 2020. Improving Text Generation Through Introducing Coherence Metrics. Cybernetics and Systems Analysis, 56(1), pp.13-21.

[17] J.-W. Lin, J.-H. Tseng, and R.-G. Chang, "Chinese Story Generation Using Conditional Generative Adversarial Network," 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), 2020.

[18] Y. Yin, D. Feng, Y. Li, S. Yin, and Z. Shi, "Text prediction method based on multi-label attributes and improved maximum entropy model," Journal of Intelligent Fuzzy Systems, vol. 34, no. 2, pp. 1097–1109, 2018.

[19] T. Akimoto, S. Imabuchi, and T. Ogata, "A story generation mechanism based on the cooperation of micro/macro story techniques: As a module in the integrated narrative generation system," 2013 IEEE/ACIS 12th International Conference on Computer and Information Science (ICIS), 2013

[20] H. Trost, J. Matiasek, and M. Baroni, "The Language Component Of The Fasty Text Prediction System," Applied Artificial Intelligence, vol. 19, no. 8, pp. 743–781, 2005.