Original Article

Automatic Next Word Generation for Text based Apps using Generative Pretrained Transformer model Compared with N-gram model

P. Niharika[1], S. John Justin Thangaraj[2]

[1]Research Scholar, Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, Tamilnadu, India, Pincode: 602105.
[2]Project Guide, Corresponding Author, Department of Computer Science and Engineering Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, Tamilnadu, India, Pincode: 602105.

Abstract

Aim: The aim of this paper is to implement automatic next word generation for text based apps using novel generative pretrained transformers and improving the accuracy in comparison with n-gram approach.
Materials and Methods: N-gram and generative pretrained transformer models are applied on data, text file that consists of a sequence of words. N-gram models accuracy of the next word which compares generative pretrained transformer models. GPT-2 has been proposed and developed. The sample size was measured as per group 9876 with G-power value 0.8.
Results: The accuracy was maximum in next word generation for text based apps using novel generative pretrained transformers 89.23% with minimum mean error when compared with N-gram model for the same dataset with p value 0.02 ($p<0.05$).
Conclusion: The study proves that novel generative pretrained transformers exhibit better accuracy than N-gram in suggesting the next word for text based apps.

Keywords: Novel Generative Pretrained Transformer, N-Gram, Tokenization, Next Word Generation, Language Model, Natural Language Processing.

DOI: 10.47750/pnr.2022.13.S04.094

INTRODUCTION

Novel generative pretrained transformer-2 is used to create text. AI that is open to all novel generative pretrained transformer-2 is a transformer-based, autoregressive language model that performs well on a variety of tasks, including (long form) text production. The basic task of anticipating the next word was used to train novel generative pretrained transformer-2 on 40GB of high-quality information (Métais et al. 2020). The model accomplishes this through the use of attention. Since the N-gram is so famous, a lot of related research has focused on using it to predict words. Only the previous n-1 words are conditioned on in an n-gram model when making a prediction (Tur and De Mori 2011). notice that this represents the length of context rather than the total number of words in the data. The trigram n-gram model, which divides the training data into three words by using tokenization, is an example of a typical n-gram model (Yildirim and Asgari-Chenaghlu 2021). The evidence for predicting the last word was based on the first two words. The application of this research includes mobile applications related to the word processing and online text editor tools ((Irene et al. 2021; Beulah et al. 2021)). The main application areas where natural language processing is employed are speech recognition, language translation, question answering, language generation and summarization.
In the last 5 years, the google scholar has published more than 196 articles and the IEEE published more than 200 articles related to the word prediction models. Language models are used informally to assess the likelihood of a sequence of words. Using LM and natural language processing , it is possible to estimate the probability of the sequence P(W1,W2,...,wm) given a sequence of words using tokenization of length m (Brownlee 2018). Different sorts of LM exist depending on the situation. During the first phase of the experiment, each user created an LM employing the user's and another of her selected persons talking data by natural language processing. The original language model was made by User1 using the interaction with person 104, while the initial Language Model was formed by User2 using the interaction with person 203 (Ardissono, Brna, and Mitrovic 2005; Seargeant and Tagg

2014). When producing the LM , our application also offered Laplace (Add-One) smoothing. Because the Language Model was being built on a single, brief interaction document, smoothing and tokenization was required (Irene et al. 2021; Beulah et al. 2021). By permuting the sampling origin of a word, 200 sets of independent and identically distributed experimental data (generated from independent and identically distributed random variables) can be obtained (Irene et al. 2021). Each sample thus acquired can be used to calculate the probability and its confidence limits. The data relative independence is the result of a period of relatively large sampling and a similar distribution derived from the stationary hypothesis postulated for natural language. This criterion of stationarity is embedded in a more general hypothesis that multiple ergodic Markov chains can represent natural languages well (Mitzenmacher and Upfal 2005). It is possible to pick and insert it into the suggested words (Bird, Klein, and Loper 2009) .With a simple tap, you may access the text. Of course, the best case scenario is the suggestion's appearance of the desired term. After as few written characters as feasible, create a list.Our team has extensive knowledge and research experience that has translate into high quality publications(Bhansali et al. 2021; Jayanth et al. 2021; Sudhakar, Ravel, and Perumal 2021; Sathiyamoorthi et al. 2021; Deepanraj et al. 2021; Raju et al. 2021; Arun Prakash et al. 2020; Kamath et al. 2020; Shanmugam et al. 2021; Rajasekaran et al. 2020; Adhinarayanan et al. 2020; Rajesh et al. 2020; Aurtherson et al. 2021)

The research gap identified from the literature is that existing approaches for the larger datasets have poor accuracy. So to increase the accuracy for the next word prediction uses various approaches for the next word generation. The study's goal is to increase next word generation accuracy by combining novel generative pretrained transformer-2 and comparing it to n-gram models.

MATERIALS AND METHODS

The proposed work is done in the computer networks lab, Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences. The sample size was measured as per group 9876 with G-power value 0.8. The accuracy in predicting the next word was performed by evaluating two groups. The study uses a dataset downloaded from gutenberg website. This study was developed using jupyter notebook software,and hardware configurations are intel i5 core processor, 50GB HDD, 4GB RAM, and the software configurations are windows OS, python jupyter notebook. The work was carried out on 9876 records from the text file from an online website gutenberg. The independent variable is the text file given as an input to the prediction model and the dependent variable is the prediction output (Tachibana and Otsuka 2018).

Generative Pretrained Transformer Model

Novel generative pretrained transformer-2 is a large transformer-based language model trained on a dataset of 8 million web pages. It's objective is to predict the next word, based on all the previous words within some text. Use the Hugging Face Transformer library which provides over 32+ pretrained models for NLG and NLU (ready to use in PyTorch and TensorFlow 2.0).

Pseudocode for Generative Pretrained Transformer Model

Step 1: install Transformers and import useful libraries
Step 2: Load datasets
Step 3: Text Tokenization
Step 4: Trainer class
Step 5: Text generation
Step 6: Text generation with different decoding methods

1. text_beam
2. text_random_sampling
3. text_k_sampling
4. text_p_sampling

N-gram Model

An N-gram model is built by counting how often word sequences occur in corpus text and then estimating the probabilities, since a simple N-gram model has limitations, improvements are often made via smoothing, interpolation and backoff.

Pseudocode for N-gram Model

Step 1:Import libraries and package
Step 2: Remove unnecessary data
#remove punctuations and make the string lowercase
#changes the word to lowercase and removes punctuations from it
Step 3: Do processing

Step 4:Load the corpus for the dataset
Step 5: create the different Nc dictionaries for ngrams
#create trigram probability dictionary
#create bigram probability dictionary
#sort the probability dictionaries of quad,tri and bi grams
Step 6:Take user input and print output

Statistical Analysis
The SPSS statistical software was used in the research for statistical analysis. Using dependent variables like training examples, corpus values, vocabulary size, input sequence length and independent variables are accuracy and loss values (Kandasamy et al. 2021). Group statistics and independent sample tests were performed on the experimental results and the graph was built for two graphs with two parameters under the study.

RESULTS

The proposed algorithm GPT-2 and existing algorithm N-gram model were run at a time in jupyter notebook. As the sample sets are executed for a number of iterations the accuracy and loss values of GPT-2 and N-gram classifiers vary for prediction of next word as shown in Table 1. Analysis of overall prediction of next word by GPT-2 and N-gram model is done. GPt-2 shows better accuracy 89.23 % than N-gram. The statistical analysis for the parameters of accuracy and loss based on the iterations and epochs were done. The standard error is also less in GPT-2 in comparison with N-gram as shown in Table 2. In Figure 1, the comparison of the significance level for GPT-2 and N-gram models was analyzed with the value P=0.01, Both GPT-2 and N-gram have a less significant level less than 0.05 .

DISCUSSION

The Context Based Word Prediction system outperforms the standard frequency-based technique. Various Markov models were examined in order to determine which one better modeled the causal relationship in between two parameters For sequence tagging, the SVMHMM model was used (Zucchini, MacDonald, and Langrock 2017). Due to the fact that it was proven to be improper for the given problem. There are numerous classes. The bi-gram model can be extended to a tri-gram or more, but as SMS text messages are usually short, it's not necessary. A greater N-gram model would be useless for short sentences (Hasida and Pa 2018). At the moment, simply model first-order Markov dependency between the position of subsequent words.

As previously stated, when the network's depth is raised without special constraints, duplicate network layers are created, resulting in poor network performance (Long and Sedghi 2019). This study uses the residual link to the original MGU to overcome the problem of network deterioration (Tachibana and Otsuka 2018). In addition, the activation function of the candidate hidden state is changed to the ReLU activation function, which avoids the vanishing gradient induced by the saturation function and allows for the training of a deeper network . Train it on another supervised corpus during fine tuning, supervised in the sense that consider the words in the history and the words to be predicted next given that history. Many people have employed GPT-1 since it has fewer parameters than GPT-2 and GPT-3, making it ideal for low-memory environments (Ramachandran, Liu, and Le 2017). Study proved that initializing seq2seq models with pre-trained language models as encoders and decoders benefits the model
(Irene et al. 2021; Tachibana and Otsuka 2018; Zucchini, MacDonald, and Langrock 2017). Their research introduces the MCNN-ReMGU model for natural language word prediction, which is based on multi-window convolution and residual-connected MGU network combined with data normalization technology. Demonstrate the effectiveness of multi-window convolution and residual-connected MGU networks in extracting high-dimensional features between relatively adjacent words and feature information between word sequences, respectively, using the PTB dataset. GPT-1 is a decoder-only transformer that predicts following words based on probability using masked self attention. GPT-1 was trained using data with a vocabulary size of 40478 and a maximum sequence length of 512 characters (Irene et al. 2021). It creates a model of language. To construct a language model, an Innovative Generative pretrained transformer is pretrained on an unsupervised corpus.
The limitations of the proposed language model do not have required memory for storing the predicted words .It works well for input data files of size less than 40 Mb. For larger data files, more memory and CPU power will be needed. Although the proposed methodology obtained satisfactory results, the approach's shortcoming is the requirement for enhanced word prediction accuracy and memory for storing the predicted word. This could be paired with more data text files in the future, resulting in improved outcomes.

CONCLUSION

The results show that the proposed Innovative Generative pretrained transformer model outperforms N-gram in terms of accuracy and loss for next word generation. The proposed Generative pretrained transformer model proves with better accuracy 89.23% when compared with N-gram for next word generation.

Declarations

Conflict of Interests

No conflict of Interest in this manuscript.

Authors Contributions

Author PNR was involved in data collection, data analysis, implementation, algorithm framing and manuscript writing. Author JJT was involved in designing the workflow, guidance and review of manuscript.

Acknowledgements

We would like to acknowledge Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (formerly known as Saveetha University) for providing facilities and constant help to carry out this study.

Funding

We thank the following organizations for providing financial support that enabled us to complete the study.

1. Qbecinfosol, Chennai.
2. Saveetha University
3. Saveetha Institute of Medical and Technical Sciences.
4. Saveetha School of Engineering.

REFERENCES

1. Adhinarayanan, Rajesh, Aravindh Ramakrishnan, Gopal Kaliyaperumal, Melvinvíctor De Poures, Rajesh Kumar Babu, and Damodharan Dillikannan. 2020. "Comparative Analysis on the Effect of 1-Decanol and Di-N-Butyl Ether as Additive with diesel/LDPE Blends in Compression Ignition Engine." Energy Sources, Part A: Recovery, Utilization, and Environmental Effects, June, 1–18.
2. Ardissono, Liliana, Paul Brna, and Antonija Mitrovic. 2005. User Modeling 2005: 10th International Conference, UM 2005, Edinburgh, Scotland, UK, July 24-29, 2005, Proceedings. Springer Science & Business Media.
3. Arun Prakash, V. R., J. Francis Xavier, G. Ramesh, T. Maridurai, K. Siva Kumar, and R. Blessing Sam Raj. 2020. "Mechanical, Thermal and Fatigue Behaviour of Surface-Treated Novel Caryota Urens Fibre–reinforced Epoxy Composite." Biomass Conversion and Biorefinery, August. https://doi.org/10.1007/s13399-020-00938-0.
4. Aurtherson, P. Babu, Bhanu Teja Nalla, Karthikeyan Srinivasan, Kulmani Mehar, and Yuvarajan Devarajan. 2021. "Biofuel Production from Novel Prunus Domestica Kernel Oil: Process Optimization Technique." Biomass Conversion and Biorefinery, May. https://doi.org/10.1007/s13399-021-01551-5.
5. Beulah, J. Rene, J. Rene Beulah, C. Pretty Diana Cyril, S. Geetha, and D. Shiny Irene. 2021. "Towards Improved Detection of Intrusions with Constraint-Based Clustering (CBC)." International Journal of Computer Networks and Applications. https://doi.org/10.22247/ijcna/2021/207980.
6. Bhansali, Karan J., Kamlesh R. Balinge, Subodh U. Raut, Shubham A. Deshmukh, M. Senthil Kumar, C. Ramesh Kumar, and Pundlik R. Bhagat. 2021. "Visible Light Assisted Sulfonic Acid-Functionalized Porphyrin Comprising Benzimidazolium Moiety for Photocatalytic Transesterification of Castor Oil." Fuel 304 (November): 121490.
7. Bird, Steven, Ewan Klein, and Edward Loper. 2009. Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit. "O'Reilly Media, Inc."
8. Brownlee, Jason. 2018. Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python. Machine Learning Mastery.
9. Deepanraj, B., N. Senthilkumar, D. Mala, and A. Sathiamourthy. 2021. "Cashew Nut Shell Liquid as Alternate Fuel for CI Engine—optimization Approach for Performance Improvement." Biomass Conversion and Biorefinery, February. https://doi.org/10.1007/s13399-021-01312-4.
10. Hasida, Kôiti, and Win Pa Pa. 2018. Computational Linguistics: 15th International Conference of the Pacific Association for Computational Linguistics, PACLING 2017, Yangon, Myanmar, August 16–18, 2017, Revised Selected Papers. Springer.
11. Irene, D. Shiny, D. Shiny Irene, V. Surya, D. Kavitha, R. Shankar, and S. John Justin Thangaraj. 2021. "An Intellectual Methodology for Secure Health Record Mining and Risk Forecasting Using Clustering and Graph-Based Classification." Journal of Circuits, Systems and Computers. https://doi.org/10.1142/s0218126621501358.
12. Jayanth, Bellappu Venkat, Melvin Victor Depoures, Gopal Kaliyaperumal, Damodharan Dillikannan, Dilipsingh Jawahar, Kumaran Palani, and Ganesha Prasad Meravanigee Shivappa. 2021. "A Comprehensive Study on the Effects of Multiple Injection Strategies and Exhaust Gas Recirculation on Diesel Engine Characteristics That Utilize Waste High Density Polyethylene Oil." Energy Sources, Part A: Recovery, Utilization, and Environmental Effects, June, 1–18.
13. Kamath, Manjunath, Subha Krishna Rao, Jaison, Sridhar, Kasthuri, Gopinath, Sivaperumal, and Shantanu Patil. 2020. "Melatonin Delivery from PCL Scaffold Enhances Glycosaminoglycans Deposition in Human Chondrocytes – Bioactive Scaffold Model for Cartilage Regeneration." Process Biochemistry 99 (December): 36–47.
14. Kandasamy, Venkatachalam, Pavel Trojovský, Fadi Al Machot, Kyandoghere Kyamakya, Nebojsa Bacanin, Sameh Askar, and Mohamed Abouhawwash. 2021. "Sentimental Analysis of COVID-19 Related Messages in Social Networks by Involving an N-Gram Stacked Autoencoder Integrated in an Ensemble Learning Scheme." Sensors 21 (22). https://doi.org/10.3390/s21227582.
15. Long, Philip M., and Hanie Sedghi. 2019. "On the Effect of the Activation Function on the Distribution of Hidden Nodes in a Deep Network." Neural Computation. https://doi.org/10.1162/neco_a_01235.
16. Métais, Elisabeth, Farid Meziane, Helmut Horacek, and Philipp Cimiano. 2020. Natural Language Processing and Information Systems: 25th International Conference on Applications of Natural Language to Information Systems, NLDB 2020, Saarbrücken, Germany, June 24–26, 2020, Proceedings. Springer.

17. Mitzenmacher, Michael, and Eli Upfal. 2005. Probability and Computing: Randomized Algorithms and Probabilistic Analysis. Cambridge University Press.
18. Rajasekaran, S., D. Damodharan, K. Gopal, B. Rajesh Kumar, and Melvin Victor De Poures. 2020. "Collective Influence of 1-Decanol Addition, Injection Pressure and EGR on Diesel Engine Characteristics Fueled with diesel/LDPE Oil Blends." Fuel 277 (October): 118166.
19. Rajesh, A., K. Gopal, De Poures Melvin Victor, B. Rajesh Kumar, A. P. Sathiyagnanam, and D. Damodharan. 2020. "Effect of Anisole Addition to Waste Cooking Oil Methyl Ester on Combustion, Emission and Performance Characteristics of a DI Diesel Engine without Any Modifications." Fuel 278 (October): 118315.
20. Raju, P., K. Raja, K. Lingadurai, T. Maridurai, and S. C. Prasanna. 2021. "Glass/Caryota Urens Hybridized Fibre-Reinforced nanoclay/SiC Toughened Epoxy Hybrid Composite: Mechanical, Drop Load Impact, Hydrophobicity and Fatigue Behaviour." Biomass Conversion and Biorefinery, March. https://doi.org/10.1007/s13399-021-01427-8.
21. Ramachandran, Prajit, Peter Liu, and Quoc Le. 2017. "Unsupervised Pretraining for Sequence to Sequence Learning." Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. https://doi.org/10.18653/v1/d17-1039.
22. Sathiyamoorthi, Ramalingam, Gomathinayakam Sankaranarayanan, Dinesh Babu Munuswamy, and Yuvarajan Devarajan. 2021. "Experimental Study of Spray Analysis for Palmarosa Biodiesel-diesel Blends in a Constant Volume Chamber." Environmental Progress & Sustainable Energy 40 (6). https://doi.org/10.1002/ep.13696.
23. Seargeant, P., and C. Tagg. 2014. The Language of Social Media: Identity and Community on the Internet. Springer.
24. Shanmugam, Rajasekaran, Damodharan Dillikannan, Gopal Kaliyaperumal, Melvin Victor De Poures, and Rajesh Kumar Babu. 2021. "A Comprehensive Study on the Effects of 1-Decanol, Compression Ratio and Exhaust Gas Recirculation on Diesel Engine Characteristics Powered with Low Density Polyethylene Oil." Energy Sources, Part A: Recovery, Utilization, and Environmental Effects 43 (23): 3064–81.
25. Sudhakar, M. P., Merlyn Ravel, and K. Perumal. 2021. "Pretreatment and Process Optimization of Bioethanol Production from Spent Biomass of Ganoderma Lucidum Using Saccharomyces Cerevisiae." Fuel 306 (December): 121680.
26. Tachibana, Kanta, and Kentaro Otsuka. 2018. "Wind Prediction Performance of Complex Neural Network with ReLU Activation Function." 2018 57th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE). https://doi.org/10.23919/sice.2018.8492660.
27. Tur, Gokhan, and Renato De Mori. 2011. Spoken Language Understanding: Systems for Extracting Semantic Information from Speech. John Wiley & Sons.
28. Yildirim, Savas, and Meysam Asgari-Chenaghlu. 2021. Mastering Transformers: Build State-of-the-Art Models from Scratch with Advanced Natural Language Processing Techniques. Packt Publishing Ltd.
29. Zucchini, Walter, Iain L. MacDonald, and Roland Langrock. 2017. Hidden Markov Models for Time Series: An Introduction Using R, Second Edition. CRC Press.

TABLES AND FIGURES

Table 1. Comparison of accuracy and loss achieved during the evaluation of GPT-2 vs n-gram for prediction with different iterations /epochs.

Algorithm	Corpus	Training examples	Test dataset	Epochs	Accuracy	Loss/Uncertainty
GPT-2	79741	12353	9876	150	60.23	40.78
				200	79.98	59.48
				280	89.23	69.45
N-gram	79741	12353	9876	2-gram	45.23	47.45
				3-gram	67.56	59.23
				4-gram	70.83	69.54

Table 2. Statistical analysis of mean,standard deviation and standard error mean of algorithms Like GPT-2 vs N-gram.

	Group	N	Mean	Std.deviation	Std.error mean
ACCURACY	1	3	76.48	14.81	8.55
	2	3	61.20	13.93	8.04
LOSS	1	3	56.57	14.55	8.40
	2	3	58.74	11.05	6.38

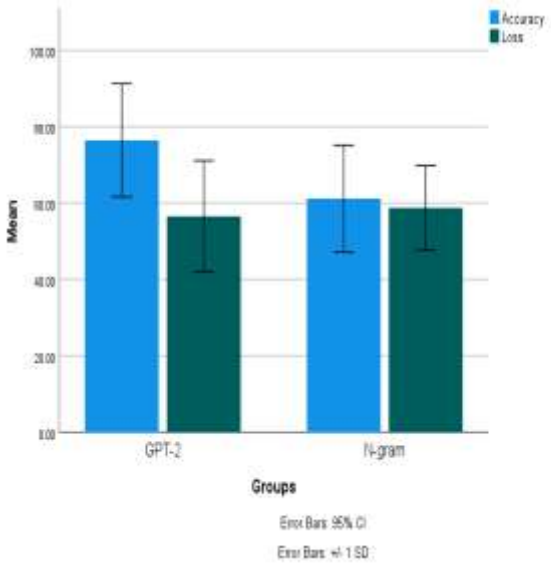

Fig. 1. Comparison of mean accuracy and mean loss of both GPT-2 and N-gram. The standard error appears i.e., +/- 1 SD to be less in GPT-2 compared to N-gram. X-axis: GPT-2 vs N-gram algorithm. Y-axis: Mean accuracy and Mean loss.