



Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.



An intelligent Chatbot using deep learning with Bidirectional RNN and attention model

Manyu Dhyani*, Rajiv Kumar

G. L. Bajaj Institute of Technology and Management, Greater Noida, Uttar Pradesh, India

ARTICLE INFO

Article history:

Received 12 April 2020

Received in revised form 15 May 2020

Accepted 16 May 2020

Available online 10 June 2020

Keywords:

Deep learning

Chatbot

Bidirectional RNN and Attention model

Tensorflow

Neural Machine Translation

ABSTRACT

This paper shows the modeling and performance in deep learning computation for an Assistant Conversational Agent (Chatbot). The utilization of Tensorflow software library, particularly Neural Machine Translation (NMT) model. Acquiring knowledge for modeling is one of the most important task and quite difficult to preprocess it. The Bidirectional Recurrent Neural Networks (BRNN) containing attention layers is used, so that input sentence with large number of tokens (or sentences with more than 20–40 words) can be replied with more appropriate conversation. The dataset used in the paper for training of model is used from Reddit. The model is developed to perform English to English translation. The main purpose of this work is to increase the perplexity and learning rate of the model and find Bleu Score for translation in same language. The experiments are conducted using Tensorflow using python 3.6. The perplexity, leaning rate, Bleu score and Average time per 1000 steps are 56.10, 0.0001, 30.16 and 4.5 respectively. One epoch is completed at 23,000 steps. The paper also study MacBook Air as a system for neural network and deep learning.

© 2019 Elsevier Ltd. All rights reserved.

Selection and peer-review under responsibility of the scientific committee of the 3rd International Conference on Science and Engineering of Materials.

1. Introduction

The Chatbot can be defined as a software which help humans to make coherent conversation with machine using natural language like English, etc. The conversation can be engaging at times with large vocabularies and broad range of conversational topics. Recently, the usage of deep learning has increased in industry and Chatbot is one of its application [1–3]. Fig. 1 shows user using the Chatbot for its various application. This paper will help to create open-domain Chatbot, which can be later subjected to a particular domain, if needed as shown in Fig. 1. It can be done by making changes in dataset, which means training model with particular domain knowledge. Due to open domain nature of the Chatbot, it can be used in making Artificial Intelligence Assistant which can make real life conversation with its user in any topic and situation. To make deep learning utilized by everyone, a major deep learning library Tensorflow is implemented by Google [4] and made available for use as an open source. Tensorflow [5] is Python-friendly library bundled with machine learning and deep learning (neural

network) models and algorithms. The paper shows the formation of Chatbot by Neural Machine Translation (NMT) model which is improvement on sequence-to-sequence model. The Chatbot use Bidirectional Recurrent Neural Network (BRNN) [6]. The BRNN was chosen, as conversation or input to the Chatbot is dynamic, which means the length of input is unfixed. The BRNN is also supported by attention mechanism [7,8], which help to further increase capacity of model to remember longer sequence of sentences. The concept of Bidirectional Recurrent Neural Network, can be understand by taking two independent Recurrent Neural Network (RNN) [9] together, sending signals through their layer in opposite directions. So BRNN can be seen as neural network connecting two hidden layers in opposite directions to a single output. This helps the network to have both forward and backward information at every step, i.e. to receive information from both past and future states. The input is fed in one direction in normal time order, and the other, in reverse order. The concept of Extended Long Short Term Memory (ELSTM) [10] can also be used, with Dependent BRNN (DBRNN), as it help to increase the result by 30% on labeled data. The training of the BRNN is done in a same way as RNN, as two bidirectional neurons do not interact with one another. When

* Corresponding author.

E-mail address: manyudhyani@bajajit.com (M. Dhyani).

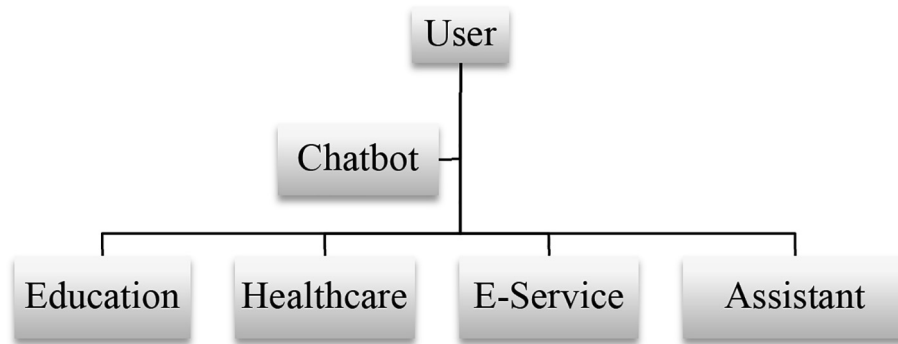


Fig. 1. Chatbot applications across various domains.

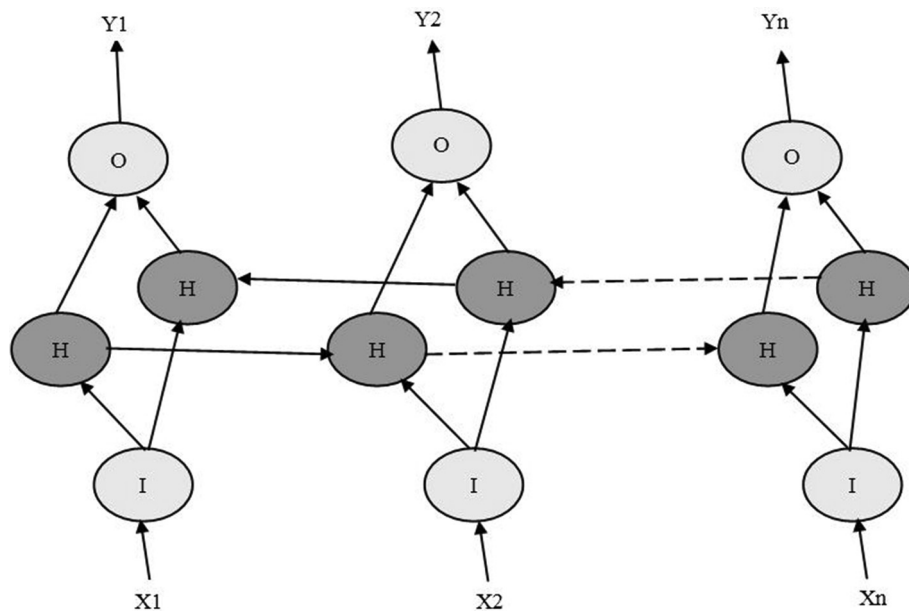


Fig. 2. Basic Bidirectional Recurrent Neural Network Architecture.

forward and backward passes are done [11], then only weights are updated.

The Fig. 2 shows the general BRNN architecture with the hidden states for forward and backward direction. The variable 'O', 'I' and 'H' means 'Output', 'Input' and 'Hidden' states respectively. The values $\{X_1, X_2, \dots, X_n\}$ are input signals to the network and values $\{Y_1, Y_2, \dots, Y_n\}$ are computed output signals from the network.

1.1. System details

This paper also evaluates MacBook as a system for deep learning. Table 1 shows the system specification and other software

Table 1
System and Software Specification.

System	Specification
CPU	Intel(R) Core(TM) i5-5350U CPU@1.8GHz • RAM: 8GB 1600 MHz DDR3
GPU	Intel HD Graphics 6000 1536MB • 48 @300-1000(Boost)MHz
Software	Os: MacOS Majave (64-bits) Version: 10.14.6 • Tensorflow Version: 1.15.0rc1

details like operating system and version of TensorFlow used. Also the technology is getting upgraded every day, even if we take Central Processing Unit (CPU) and Graphics Processing Unit (GPU), which are becoming faster [12]. The system used is 2017 MacBook Air series by Apple. The laptop was at room temperature all the time of training the model.

2. Literature review

There have been increase in development of conversational agent systems in commercial market like in retail, banking and education sectors. The research is being done, to improve accuracy and make conversation between Chatbot and user as close to real world conversations. Apart from traditional rule based technique, used earlier in Chatbot development and other straightforward machine learning algorithm, advance concepts and techniques like Natural Language Processing (NLP) techniques and Deep Learning Techniques like Deep Neural Network (DNN) and Deep Reinforcement Learning (DRL) are being used to model and train the Chatbot systems. In early days, the translation was done by breaking up source sentences into multiple token or chunks and then translated phrase-by-phrase. Sequence-to-Sequence has been a popular model based on neural

network and deep learning, used in Neural Machine Translation [13]. It is used in tasks like machine translation, speech recognition, and text summarization. Sequence-to-Sequence model has an encoder and a decoder, both having Vanilla RNN [14] by default. The encoder take source sentence as input to build a thought vector. A thought vector is a vector space containing sequence of numbers which represent meaning of the sentence. Then finally, decoder process the thought vector fed to it and emit a translation called a target sequence or sentence. But Vanilla RNN fails when long sequence of sentences are fed to model, as information needs to be remembered. This information frequently becomes larger for bigger datasets and create bottleneck for RNN networks. The variation in RNN has to be used like BRNN with Attention mechanism or Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) [15] to handle failure in longer sequences. To understand meaning behind the sentence, the intention, facts and emotions described in the sentence, it must be analyzed. In [3], the statistical difference was 63.33% to identify two groups or personalities on basis of their sentiments. The technique of NLP and machine learning helps to deeply analyze the sentence sentiments and make comfortable environment for humans to make conversation with machine. If dataset is in text, sentiment features can be classified as text level or document level [16]. The deep learning methods like Convolutional Neural Network (CNN) [17] and RNN are used in document sentiment classification, as it is the difficult one out of the two above mentioned classification. Also developing Chatbot for young adolescent, engaging them in their most preferred channel of communication, that are smartphones and successfully helping them to adult focused care. In [2], the engagement time was 97%. The Kyoto-NMT is an open-source implementation of NMT paradigm [18]. It used chained, a Deep learning Framework. It has used two layer of LSTM with Attention model in its Recurrent Neural Network. It has also used whitespaces for making token in data preparation. The vector size used by it is of 1000. The training data is a sentence-aligned parallel corpus that is in utf-8 text files: one for source language sentences and the other for target language sentences. In this, a JSON file is created, containing all the parameters. SQLite is used for database creation and for keeping track of training progress. The bleu score is computed by applying greedy search on validation dataset. As the real translation was there (Japanese-to-English), the bleu score they reached is of 26.22. The Snowbot is a Chatbot, developed on Tensorflow and MXNet [19]. The Chatbot uses Cornell movie Dialog Corpus with 221,282 QA, size 22 MB and twitter chat with 377,265 QA having emoji in sentences, size 51 MB. The Chatbot was developed in desktop having NVIDIA GTX 1080 8 GB and 5 GB RAM. The neural network contains two layer of LSTM with 256 hidden units. The vocabulary size was of 50000. The model completed 7 epochs in 1 h. The perplexity was 90 and 135 for Cornell and Twitter dataset respectively.

2.1. Chatbot performance metrics across industries

Currently, there are many performance metrics, and certain measurement standards are followed across industry for Chatbot [20]. Different organizations need Chatbot according to the nature of their work and market surrounding it. One of the most important performance metric for Chatbot is the structure and the length of its conversation. The length of output sentence must be appropriate and in context to the conversation being done. Shorter and simple the structure of sentence in output, faster the solution, does increase the customer satisfaction rate. To understand this metric with an example of banking sector, in banks, the Chatbot is mainly used to guide the user through the bank's policy, schemes and other customer inquiries about their account. This serve user to perform their tasks quicker, and also lower the human call assistance, thus cutting cost in the service. The consumer satisfaction

also bring second metric—the retention rate, very important for Chabot success. The companies aim for significantly high retention rate, indicating customer satisfaction. The automated calls and Chatbot messengers are being used to replace other communication mediums (i.e. lowering call volumes by humans). The retention rate increases when Chatbot are more trained to support the user in managing their account without speaking to a human assistant. Another metric is the ability of the Chatbot, to produce the personalized reply to the user. This means that the Chatbot should take in the source sentence, understand and analyze it, and produce an output statement mapped to the particular problem or query of the user. The companies try to personalize and customize the output statement according to each user's need, like bank suggesting user a relevant offer or credit card scheme according to the balance in the user's account, salary, current loan and its spending history. For example, Erica, which is a Chatbot, is Bank of America's AI virtual assistant, which combine predictive analysis and NLP, to help its users to access their balance information, tips on better saving the money depending upon their spending habits, making transaction between accounts and schedule meetings at financial centers and banks. The e-market and Retail Chatbots make engaging environment for users to shop. Through their environment, the Chabot transform itself in a personal assistant for assisting in shopping. Unlike banking and financial sector Chatbots, Retail conversational agents are designed to look for higher number of conversational steps by holding the users attention, providing details and encouraging them to browse more and ultimately purchase the product. For instance, Ebay's ShopBot, help users to find best deals from its list of billion products. It is easy to-talk-to, like a friend, either if one is searching for a specific product or browsing to find something new. The above discussed studies shows network designed for small sized datasets and for short input sentences which are not fit for real life conversation as human tends to speak in longer sentences. To counter real world conversation, model like BRNN is important to know conversation context and references, from past as well as future. Attention mechanism is important attachment to the network as it help to weigh the particular references from the input sentences. Also for evaluating Chatbot performances there are no hard metrics [21], but parameters like Perplexity, Learning rate and Bleu score can represent as to how close one can approach at the time of training the model.

3. Methodology

The model, BRNN with Attention model not only help in short but also in longer tokens. Attention model help us to remember longer sequences at a time and also help in context problem where both historical and future information is required. In real world as language may not necessarily be in perfect sequence, sometimes one has to use context, hear full conversation before going back and responding to words leading up to that point. Human tends to speak in longer sentences to understand the meaning. This is the reason that makes combination of BRNN and Attention model, perfectly right choice for Chatbots. The BRNN structure forms the acyclic graph as can be seen in Fig. 3. The signals are of two types: forward and backward. The \vec{a} is forward recurrent component and \bar{a} is backward recurrent component. There are input values $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \dots, \mathbf{X}^n\}$ at timestamp $t = \{1, 2, \dots, n\}$ and predicted values are $\hat{\mathbf{Y}} = \{\hat{\mathbf{Y}}^1, \hat{\mathbf{Y}}^2, \dots, \hat{\mathbf{Y}}^n\}$. To make prediction $\hat{\mathbf{Y}}$ at t , which is an activation function $\mathbf{g}(\mathbf{n})$, computed as:

$$\hat{\mathbf{Y}} = \mathbf{g}\left(\mathbf{W}_y \left[\vec{a}^t, \bar{a}^t \right] + \mathbf{P}_y\right) \quad (1)$$

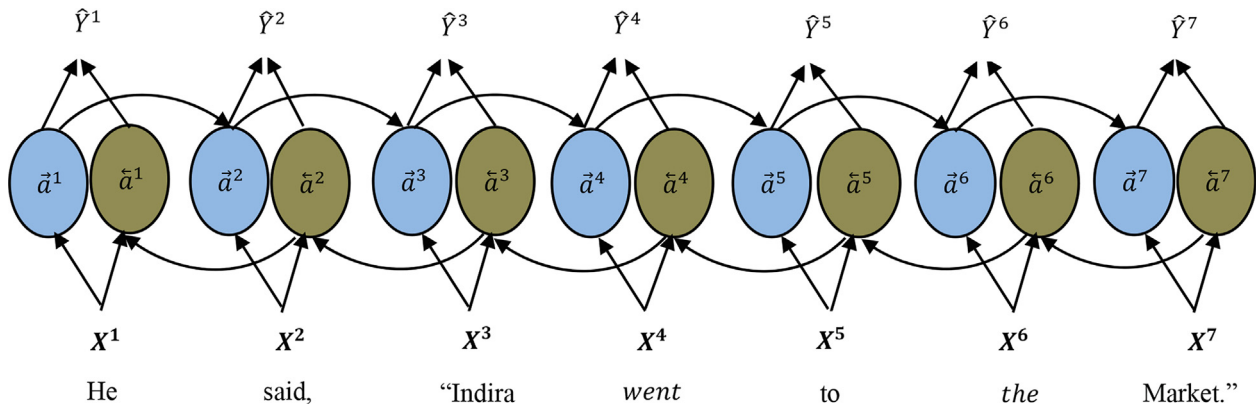


Fig. 3. Source input demonstration into BRNN with hidden layer to describe forward and backward flow of information.

where \mathbf{W}_y is the weight according to the input and magnitude set in the network, \vec{a}^t & \tilde{a}^t are the forward activation at time t and backward activation at time t respectively. The \hat{P}_y is the computed value (or predicted value) from the previous neuron in the direction, information is advancing. The activation function in neural network are the function attached to the node (or neuron), which get triggered when the input value in the current node is relevant in making prediction. There are many types of activation functions, but with BRNN, sigmoid and logistic activation functions are mostly used. For example, in the network shown in Fig. 3, an input of couple of statements has been given. Statement 1- He said, "Indira went to the market." and statement 2- He said, "Indira Gandhi was a great Prime Minister." At an instance, when statement 2 was inserted into the network, at time step 3, where word "Indira" was inserted, \hat{Y}^3 cell prediction or output signal is checked, and due to bidirectional in nature, the forward information flows from cell 1st and 2nd to 3rd cell, as well as, backward flow of information from 9th cell (through all cells in between) to 3rd, help the cell to predict that 'Indira' in statement 2 is Prime Minister and not 'Indira' who went to the market, in statement 1. If simple RNN have been used, the output must be according to statement 1, as the network didn't have the future information.

This is because the RNN are unidirectional, i.e. with positive direction of time [22]. In attention mechanism, attention vector is generated by calculating score and then calculated vector is retained in memory, so as to choose between best candidate vectors. The score is calculated by comparing each hidden target with source hidden state. For applying attention mechanism, single directional RNN can be used above BRNN structure. Each cell in the Attention network is given context, as an input. This type of network is also called context-aware attention network [23]. The predicted value from each BRNN cell is taken, combined with value from the previous state (or neuron) of the attention network for calculating the attention value. One can also say that context is weight of the features from different timestamp. The Fig. 4 shows how the weighted value is taken in attention cell, as described above. The context \mathbf{C} for cell 1 in attention network can be computed as:

$$\mathbf{C}_1 = \sum_t \mathbf{Y}^{<1,t>} \mathbf{a}^{<t>} \quad (2)$$

where $\mathbf{a}^{<t>} = (\vec{a}^t, \tilde{a}^t)$ and $\mathbf{Y}^{<1,t>}$ is the value from activation function applied on BRNN for prediction on its each cell, that will be used as weight for context computation. $\mathbf{Y}^{<t,t'>}$ is basically the amount of attention, $\mathbf{Y}^{<t'>}$ should pay to $\mathbf{a}^{<t'>}$.

The \mathbf{S}^n represents the states of the attention model, where $\mathbf{n} \in \mathbf{W}$. The \mathbf{S}^0 is the primary weight, whose value is set according to the network. The $\hat{\mathbf{Z}}^t$ at time step t is the attention value after computation from attention neuron where $t \in \mathbf{N}$. The Fig. 4 is the extended structure, which when combined with BRNN in Fig. 3 to produce complete architecture for the deep learning network. This network has quadratic cost for time. As generally, in Chatbot, not much longer sentences or paragraph are used to converse, so cost may be acceptable. Though there are other research in this field to reduce quadratic time cost. The attention mechanism is one of the important methods in deep learning techniques, especially in area of document classification [24], speech recognition [25], and in image processing [26–28].

3.1. Implementation

The procedure for implementing methodology is depicted in Fig. 5.

3.1.1. Datasets

The Reddit dataset [29] has been used to make database for the Chatbot. The dataset contain comments of January, year 2015. The format of data is in JSON format. The content of dataset is parent_id, comment_body, score, subreddit, etc. The score is most useful to set the acceptable data criteria as this show that this particular comment is most accurate reply to the parent_comment or parent_body. Subreddit can be used to make some specific type of Chatbot like scientific or other particular domain Bots. A subreddit is a specific online community, and the posts associated with it are dedicated to a particular topic that people write about. The database formed after pre-processing the dataset have size of 2.42 GB. The database contain 10,935,217 rows (i.e. number of parent comment-reply comment pairs).

3.1.2. Preprocessing

Now first, for training the model, database is required. So dataset is converted into a database with fields like parent_id, parent, comment, Subreddit, score and UNIX (to track time). To make data more admissible, take data (comment) which have less than 50 words but more than 1 word (in case reply to parent is empty). Also remove all newline character, '[deleted]' and '[removed]' comments, etc. If data (comment body) is valid according to acceptable criteria and has more score than previously paired comment to parent comment of same parent_id, then replace it. Also if encountered with a comment with no parent comment then it means, it can itself be parent comment to some other comment (i.e., it is main thread comment in Reddit). For database creation, the data

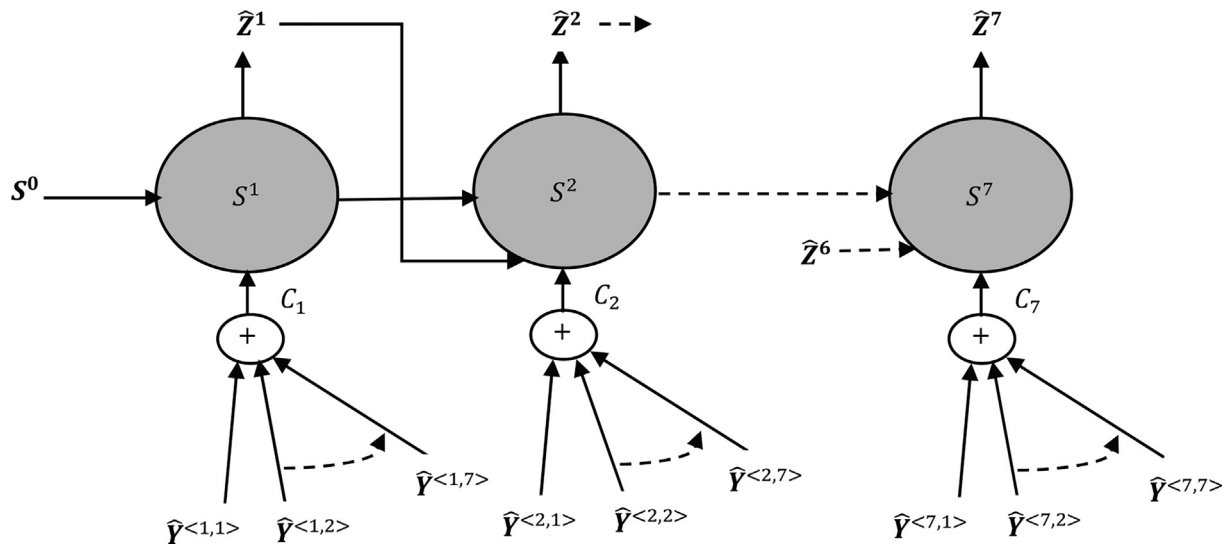


Fig. 4. Context-Aware Attention Model architecture.

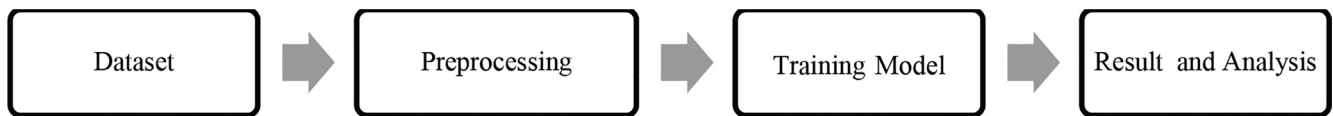


Fig. 5. Procedure for implementing methodology.

have been paired in parent and child comments. Each comment is either a main parent comment or reply comment, but each have parent_id. Each parent comment and its reply comment has same parent_id. The pairs are made in accordance to the parent_id. In creation of database, the parent comment is mapped with its best child or reply comment. Any comment, either a parent or a child, have an acceptance score of two. When encountered with a new comment, if it matches the parent_id of previous entered reply comment to a parent body, then compare it with entered reply comment score. If current comment has better score than existing mapped reply comment, the replacement is done between new and previous reply comment and other associated data. If not the case, then the row remains unchanged. Further, if comment encountered has a parent body which is not yet paired with any reply comment, map the comment with its parent body, else if comment has no parent body, then create a new row for the comment, as the new comment encountered can be a parent to some other reply comment. On creation of database, 10,935,217 parent-reply comment pairs (rows) are created.

3.1.3. Training model

For training after creation of database, rows have to be divided into training data and test data. For both, two files are created (i.e., Parent comment and Reply comment). Training data contains 3,027,254 pairs and Test data contains 5100 pairs. There are also list of protected phrases (e.g.www.xyz.com should be a single token) and blacklisted words, to avoid feeding it to learning network. The training files are fed to multiprocessing tokenizer, as they are CPU intensive. The sentences will be divided into tokens on basis of space and punctuation. Each token will act as vocabulary. For each step, vocabulary size is 15000. The size is appropriate for systems having virtual memory of 4 Gigabytes. The RegX module is used for formulating search pattern for vocabulary. It is faster than standard library and it is basically used to check whether a string contain a specific search pattern. The neural network must

be designed as mentioned above. Once the training starts, the main concerned hyperparameters (HParams) in metrics are bleu score (bleu), perplexity (ppl) and learning rate (Lr). Bleu score tells, how good the model is translating a sentence from one language to another language. It should be as high as possible. Perplexity is a measure of the probability distribution, or it tells about model prediction error. Learning rate reflects the model's learning progress in the network. As in this paper, language at both ends of the model is English, so Perplexity is more useful than bleu score. Learning rate is useful but only when model is trained with large data and for longer period of time. If model is trained for limited period of time or with less data, no significant change in learning rate will be observed.

3.1.4. Result and analysis

Initially, the perplexity before training the model was 16322.15, learning rate was 0.001 and bleu score was 0.00. The average time used by above described system, per 1000 steps is between 4 and 4.5 h. If upper bound of time is taken, then for training machine till 23,000 steps, the system took 103.5 h. The perplexity, learning rate and bleu rate at step 23,000 is 56.10, 0.0001 and 21.67. The maximum value of bleu score model reached was of 30.16 at 18000thstep. The model also passed one epoch at 23000th step. The learning rate is low and negligible, as changes were made externally to the weights in the neural network, once the training started. The performance evaluation is shown in Table 2.

Table 2
Performance evaluation.

Parameter	Initial	After training 23,000 steps
Perplexity	16322.15	56.10
Learning rate	0.001	0.0001
Bleu score	0.00	21.67
Average time (per 1000 steps)	0	4.5 h per 1000 steps

Table 3
Performance comparison.

Ref.	Parameter	Ref. Domain	Our Domain	Ref. Result	Our Result
[18]	Bleu score	Japanese-to-English	English-to-English	22.86	30.16
[19]	Perplexity	Cornell Dataset (22 MB)	Reddit Dataset (2.42 GB)	90	56.10
[19]	Perplexity	Twitter Dataset (51 MB)	Reddit Dataset (2.42 GB)	135	56.10
[19]	Time	83.7 h for 1,000,000 steps	103.5 h for 23,000 steps	10 epoch	1 epoch

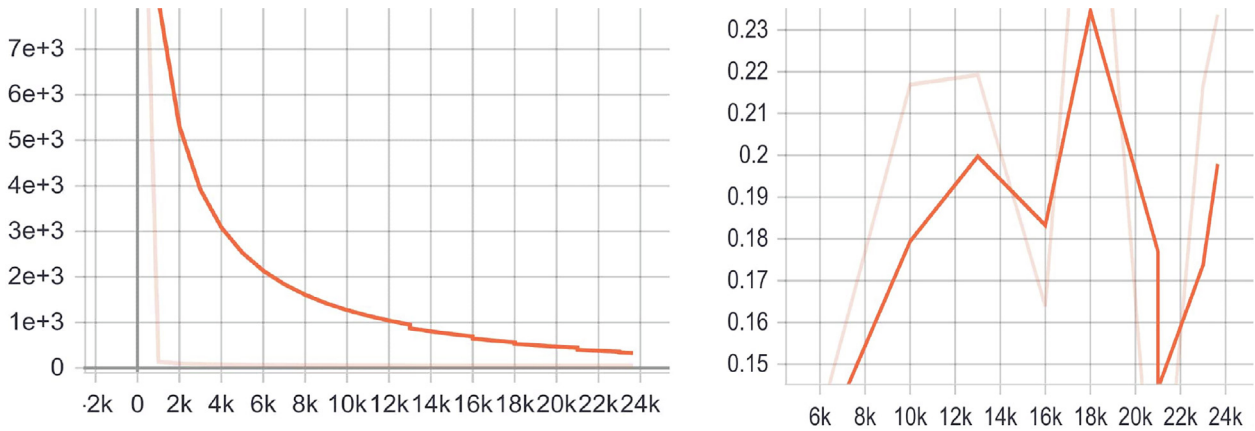


Fig. 6. The Perplexity and Bleu Score graph.

Also one can draw comparison of the performances of the Chatbot in [18] and in [19] with our result, as reflected in Table 3.

The graph of perplexity and bleu score is shown in Fig. 6. The other observations in Tensorboard like train_loss, decreases when model starts training, and once if train_loss starts increasing after reaching a minimum point, one should stop training the model, as very less or no change in model performance will occur. Thus, it describe that more and excessive training of model can lead to data loss. The smoothing of all graphs is done at value of 0.96 for better interpretation.

Speed graph in Fig. 7 demonstrate the system speed per 1000 steps. As mentioned above, no real translation is going on this Chatbot, but still value should initially increase. There will be decrease in value too at some points as no real translation is taking place. The perplexity should fall in every case. If it doesn't fall, it means the model is not getting trained properly. Also there will be not significant change in reply of NMT Chatbot. There is speed variation in value of speed of system as the speed depends upon overall task getting performed and other opened-up and running applications. From analysis and experience on system while working on experiment, the MacBook Air is just enough for basic deep

learning model training, but not adequate. If one wants to go higher, and train some intermediate and advance model, MacBook Air (2017) hardware is not enough. There are insignificant change in latest model of MacBook Air, for instance MacBook Air (2020) or other MacBook Air series.

With help of test file, previously created for validating Chatbot's reply, Table 4 gives the comparison between the source dataset's test reply comment and the Chatbot's (NMT-Chatbot) reply after 23,000 steps of training. The test reply comment means the real world human reply to the test parent comment on Reddit. The NMT-reply is the output from the Chatbot. The eight sentences are randomly picked from the parent comment field of the database created for training. The sentences below have length range between 8 and 30, in terms of words. No punctuation in the sentences shown in Table 4 are added or removed.

4. Conclusion and future works

A Chatbot using deep learning NMT model with Tensorflow has been developed. The Chatbot architecture was build-up of BRNN and attention mechanism. The Chatbot Knowledge base is open

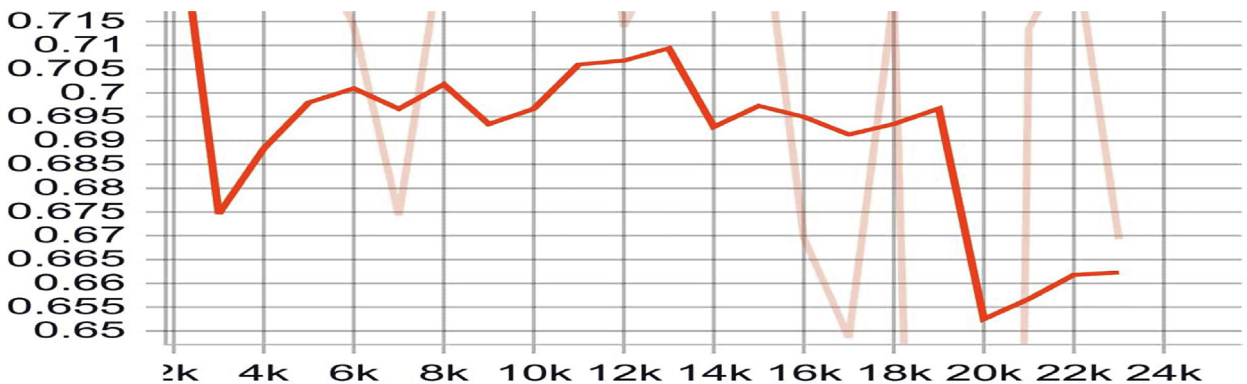


Fig. 7. The System speed graph (x-axis denotes number of steps and Y-axis denotes system time (in sec).

Table 4
Conversation input–output response analysis of referenced user versus NMT-Chatbot reply.

Test parent comment	Test reply comment	NMT-chatbot Reply
I feel like I haven't been here in years. No man, I'm serious. Gyarados is one of the worst pokemon ever, used as a trick to make inexperienced trainers evolve before allowing magikarp to reach its true power levels. I know I keep reporting them but they keep coming back!	It's okay to be nostalgic kekigeddit	I have been here yesterday I don't get it
Keep it going man, I'm really interested in seeing where this goes!	Its pretty much a Mirana Ultimate with magic resist but better and cheap too	I'm not saying it's hacking, but I'm not sure if it's hacking or not.
Well it's not like they still couldn't use a little help.	Alright, I'm working on it right now	I'm not saying they're going to make a difference. I'm not sure if I'm not alone. I'm just saying that I'm not alone.
Cos this game isn't interesting at all anymore	There's a glitch in the matrix.	I agree. Klinsky was a good guy.
you're right my fault. I was looking at games started. But still he only had 33 tackles that year that is not 'superstar' numbers. all we are is dust in the wind	Good point. I'm more interested in whether or not Blake will get a triple and Crawford will start getting shit to fall. For sure, I was agreeing with you that he wasn't the Darren Woodson we came to know and love yet. dust to dust and ashes to ashes	I'm not sure if you're talking about the game. I'm not a fan of this sub.

domain, using Reddit dataset and it's giving some genuine reply. In future, the model will be rewarded on relevant and sentiment appropriate reply. This will involve Deep Reinforcement Learning (DRL) technique. Also the methodology used in implementing and training the chatbot, can be used to train the specific domain chatbot, like scientific, healthcare, security, banking, e-market and educational domain. This approach will help building the chatbot in any domain easier and can improve the existing chatbot based on simple RNN architecture or other neural network by using attention mechanism as above. To implement domain specific chatbot (like healthcare, education, etc.), one can download specific Subreddit, of the particular domain. The future work will also include to build a healthcare Chatbot, guiding patient of diseases like COVID-19 (pandemic), Diabetes, High Blood Pressure and heart, etc. by providing information about the inquired disease, food one can eat and ways to deal with several emergency situations. This Chatbot will be powered by a recommender system too. In this paper, the novel idea was to analyze MacBook Air as a system to study and train deep neural network model. We find the MacBook air as mediocre and basic level system for deep learning. This result can help basic level students or other professionals to choose system wisely before starting with deep learning.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] Heller, Bob, Mike Proctor, Dean Mah, Lisa Jewell, and Bill Cheung. Freudbot: An investigation of Chatbot technology in distance education. In *EdMedia+ Innovate Learning*, pp. 3913–3918. Association for the Advancement of Computing in Education (AACE), 2005.
- [2] Jeremy Beaudry, Alyssa Consigli, Colleen Clark, Keith J. Robinson, *Getting ready for adult healthcare: designing a Chatbot to coach adolescents with special health needs through the transitions of care*, *J. Pediatric Nursing* 49 (2019) 85–91.
- [3] Rhio Sutoyo, Andry Chowanda, Agnes Kurniati, Rini Wongso, *Designing an emotionally realistic chatbot framework to enhance its believability with AIML and information states*, *Proc. Computer Sci.* 157 (2019) 621–628.
- [4] Mo, Young Jong, Joongheon Kim, Jong-Kook Kim, Aziz Mohaisen, and Woojoo Lee. Performance of deep learning computation with TensorFlow software library in GPU-capable multi-core computing platforms. In *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 240–242. IEEE, 2017.
- [5] Abadi, Martín, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado et al. *Tensorflow: Large-scale machine learning on heterogeneous distributed systems*. arXiv preprint arXiv:1603.04467 (2016).
- [6] Mike Schuster, Kuldip K. Paliwal, *Bidirectional recurrent neural networks*, *IEEE Trans. Signal Process.* 45 (11) (1997) 2673–2681.
- [7] Xiang Li, Wei Zhang, Qian Ding, *Understanding and improving deep learning-based rolling bearing fault diagnosis with attention mechanism*, *Signal Process.* 161 (2019) 136–154.
- [8] Niclas Ståhl, Gunnar Mathiason, Göran Falkman, Alexander Karlsson, *Using recurrent neural networks with attention for detecting problematic slab shapes in steel rolling*, *Appl. Math. Model.* 70 (2019) 365–377.
- [9] Jeffrey L. Elman, *Distributed representations, simple recurrent networks, and grammatical structure*, *Machine Learning* 7 (2–3) (1991) 195–225.
- [10] Yuanhang Su, C.-C. Jay Kuo, *On extended long short-term memory and dependent bidirectional recurrent neural network*, *Neurocomputing* 356 (2019) 151–161.
- [11] Seongwoon Jeong, Max Ferguson, Jerome P. RuiHou, HoonSohn Lynch, Kincho H. Law, *Sensor data reconstruction using bidirectional recurrent neural network with application to bridge monitoring*, *Adv. Eng. Informatics* 42 (2019) 100991.
- [12] Hyeran Jeon, Woo Hyong Lee, Sung Woo Chung, *Load unbalancing strategy for multicore embedded processors*, *IEEE Trans. Computers* 59 (10) (2010) 1434–1440.
- [13] Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. *Sequence to sequence learning with neural networks*. In *Advances in neural information processing systems*, pp. 3104–3112. 2014.
- [14] Alex Sherstinsky, *Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network*, *Physica D: Nonlinear Phenomena* 404 (2020) 132306.
- [15] Wuyan Li, Wu. Hao, Nanyang Zhu, Yongnian Jiang, Jinglu Tan, Ya Guo, *Prediction of dissolved oxygen in a fishery pond based on gated recurrent unit (GRU)*, *Information Processing Agriculture* (2020).
- [16] Fagui Liu, LaileiZheng JingzhongZheng, Cheng Chen, *Combining attention-based bidirectional gated recurrent neural network and two-dimensional convolutional neural network for document-level sentiment classification*, *Neurocomputing* 371 (2020) 39–50.
- [17] Kim, Yoon. *Convolutional neural networks for sentence classification*. arXiv preprint arXiv:1408.5882 (2014).
- [18] Cromieres, Fabien. *Kyoto-NMT: A neural machine translation implementation in chainer*. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, pp. 307–311. 2016.
- [19] Guo, Pinglei, Yusi Xiang, Yunzheng Zhang, and Weiting Zhan. *Snowbot: An empirical study of building Chatbot using seq2seq model with different machine learning framework*.
- [20] Przegalinska, Aleksandra, Leon Ciechanowski, Anna Stroz, Peter Gloor, and GrzegorzMazurek. *In bot we trust: A new methodology of Chatbot performance measures*. *Business Horizons* 62, no. 6 (2019): 785–797.
- [21] Liu, Chia-Wei, Ryan Lowe, Iulian V. Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. *How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation*. arXiv preprint arXiv:1603.08023 (2016).
- [22] Yu, Wennian, Il Yong Kim, Chris Mechefske. *Remaining useful life estimation using a bidirectional recurrent neural network based autoencoder scheme*. *Mech. Systems Signal Process.* 129 (2019): 764–780.
- [23] Li, Huayu, Martin Renqiang Min, Yong Ge, and AsimKadav. *A context-aware attention network for interactive question answering*. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 927–935. 2017.

- [24] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, Eduard Hovy, Hierarchical attention networks for document classification, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016*, pp. 1480–1489.
- [25] Chan William, Navdeep Jaitly, Quoc V. Le, Oriol Vinyals. Listen, attend and spell. arXiv preprint arXiv:1508.01211 (2015).
- [26] Hua, Yuansheng, Lichao Mou, Xiao Xiang Zhu. Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional LSTM network for multi-label aerial image classification. *ISPRS J. Photogrammetry Remote Sensing* 149 (2019): 188–199.
- [27] Xu, Kelvin, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, RuslanSalakhudinov, Rich Zemel, and YoshuaBengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pp. 2048–2057. 2015.
- [28] Kumar, Rajiv, Amresh Kumar, and Pervez Ahmed. A benchmark dataset for devnagari document recognition research. In *6th International Conference on Visualization, Imaging and Simulation (VIS'13)*, Lemesos, Cyprus, pp. 258–263. 2013.
- [29] https://www.reddit.com/r/datasets/comments/3bxlg7/i_have_every_publicly_available_reddit_comment/?st=j9udbxta&sh=69e4fee7 [Reddit dataset].