IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 4, April 2022

DOI: 10.17148/IJARCCE.2022.11449

SURVEY ON NEXT WORD PREDICTION AND PARAPHRASING USING LATENT SEMANTIC ANALYSIS

S Nithin[1], Sameer Pandit[2], Tanuja Shastri[3], Yash Joshi[4], Dr.Rashmi Amardeep[5]

[1-5]Department of Information Science and Engineering, Dayananda Sagar, Academy of Technology and Management, Bengaluru-560082

Abstract: Next Word Prediction also called Language Modelling is the task of predicting the word that comes next. It is a core problem of NLP and has several applications. The paraphrasing generation techniques help to identify or to extract/generate phrases/sentences conveying similar meanings. Next word prediction (NWP) is a major challenge in the field of natural language processing. This paper mainly talks about the Recurrent Neural Network (RNN) and introduces a more effective neural network model named LSTM for supporting next word prediction and Latent semantic analysis (LSA) is a method for evaluating a piece of text using mathematical computing and examining the link between terms in the documents, as well as between documents in the corpus for supporting the paraphrasing generation technique. These models may need a significant amount of computing work, making the model inapplicable for some sorts of applications. In conclusion, although tricky and application-dependent, Proper setting of the learning rate can reduce the lingering time of the neural network.

Keywords: Latent Semantic Analysis, Paraphrasing, Next word prediction, Natural language processing, recurrent neural network, LSTM, Character prediction.

I. INTRODUCTION:

Next word prediction (NWP) is the problem of predicting which word is likely to follow a given initial text fragment. Decision-making regarding NWP in the arena of language modeling is quite frequent in the real world such as word sense. With the advance in technology and mathematical models, it is possible to develop faster systems with more accuracy. Natural Language Generation is a branch or subfield of Natural Language Processing (NLP). NLG is occasionally confused with Natural Language Understanding (NLU), another sub-field of NLP. The human brain demands in-depth comprehension and reasoning while creating language or phrases from scratch or by utilizing a given context. A computer system's ability to construct meaningful and fluent phrases is more difficult.

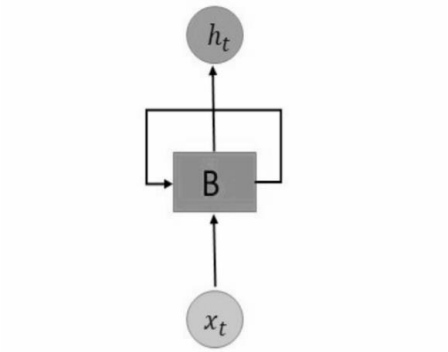

Fig. 1. A NN with some input value xt and outputs a value ht

As a result, NLG is more challenging than NLU for a system. An ideal NLG system aims toward completely replacing humans for tasks like article writing, creating summarizations quickly, real-time question answering, report generations, and streamlining operations. A more recent trend is the use of distributional semantics, where the meanings of the words are determined based on the context in which they occur. On a big scale, meaning may be retrieved from text (or speech) corpora in this manner. Contexts are represented using vectors of frequencies of other words co-occurring with a word being modeled (Lenci, 2008).

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 4, April 2022

DOI: 10.17148/IJARCCE.2022.11449

II. LITERATURE SURVEY

There are a lot of different data sets, techniques, and machine learning models that have been used by many different researchers to predict the next word using Latent semantic analysis.

H. M. Mahedi Hasan et. al**[1]** proposed that keywords are a summarization of documents that help other words in the text to inaugurate the core content. This approach is an important method for data analysis. This method's specific uses include topic modeling, key phrase extraction, document summarising, and so on. This approach is divided into two stages: pre-processing and post-processing. In, pre-processing step contains unification and removing stop words. In general, unification provides several features for processing words or texts to make them identical and useful for identifying key phrases. In a post-processing step, algorithms and approaches are used to help the proposed model extract the keywords from documents. The interior's sense of the text is technically illustrated by semantic relation, a process of finding relationships between words and phrases. The precision of advanced semantic resemblance is difficult to catch because a certain set of conditions needs to be set in order to find the accurate semantic relation. The degree of semantic relationship between two terms is determined by identifying semantic relationships and word senses among words. It is not required that the meanings of words be similar. (For example, book and paper are tightly associated with their meaning, whereas book and song are not.) The similarity between noun pairs and verb pairs is supported by WordNet, which is a semantically based technique. WordNet is a lexical chain builder that may be used to locate a synonym, meronym, hypernym, or hyponym for a given term. Quantitative and qualitative are categorized as keywords extraction techniques, where the semantic relation is used in qualitative techniques.
This model is constructed based on two steps following as pre-processing and post-processing. In the pre-processing step, the process of building a dataset is described and in the post-processing step, our main algorithm is introduced with a clear explanation. Initially, each document is divided into sentences, and the model eliminates stop words by using a list of predefined stop words, followed by stemming using the porter stemming [1] approach. POS tagging for each word and term frequency counts are incorporated into the procedure. Along with the basic features, n-grams are also used to find key phrases. But the main approach is based on the semantic relation between words at the level of the word and sentence level. The model will then establish a relationship for selected high term frequency words based on the document, which has been specified as a header and expand it with nodes. The node denotes the rest of the words and the connection between the header and nodes is based on semantic senses.
Pre-processing contains four steps in this model to make the documents ready to process with the post-processing model. In this approach, the authors have collected datasets from different books, journals, papers, etc. In the process of making the dataset, they have handled the Unicode problem in the dataset and have used regex in python programming to resolve this issue. The model eliminates stop words from the texts after processing the dataset. Some words in the texts are recognised as stop words since they do not provide any functional information to the core context. (e.g. because, between, could, would, too, the, etc.).Stemming is an automatic process to shrink prefixes and suffixes from the words, to recognize them as the same word.
At the post-processing step of this model, POS-tagging is used to tag every word with its corresponding parts of speech (e.g. ['Python', 'NNP'], [‘is’, ‘VBZ’], [‘programming’, ‘NN’]). The authors have used the natural language toolkit POS-tagging method in this model.
In this research, the model follows this particular set of conditions to make semantic relation of the text.
After pre-processing, the dataset is organized line by line with removing stop words, POS-tagging, stemming and calculating term frequency and N-grams. Later relation headers are selected based on the highest TF and selected POS-tagging terms.
In the final part of this model, the keywords extraction model is proposed based on a designed algorithm.
Headers are selected as important keywords based on their highest relation with other headers and words (nodes).
If WH exists in any header to nodes relation, then this relation is selected as an important one and extracts keywords.

Vladimir Despotovic et. al**[2]** discussed some task-specific aspects of NMF and MLN more extensively.

Multinomial naive Bayes is a variation of the naive Bayes classifier, which is commonly used in text classification. Unlike simple naive Bayes, which represents a data instance (spoken utterance) by the presence or absence of tokens (AUDs in this example), MNB represents the data instance by the number of token occurrences. The method is known for statistical language modeling for speech recognition as a unigram language model (McCallum and Nigam, 1998). The approach computes the posterior probability of the unseen test data belonging to each class in a prediction step and then assigns the observation to the class with the highest posterior probability. Support vector machines attempt binary classification by locating a decision boundary that is the point between a linearly separable collection of data that is the farthest away from any data.The hyperplane defined as the linear decision function with the maximum margin between data points belonging to various classes is the decision boundary. (Cortes and Vapnik, 1995). The support vectors reflect

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 4, April 2022

DOI: 10.17148/IJARCCE.2022.11449

a small subset of data points that lie on this margin; hence, they fully describe the hyperplane's location. If the dataset is not linearly separable, we can map training vectors di into a higher dimensional space using the transform φ(di), where the separation might be easier. We introduce the kernel function related to the transform φ(di) with the relation

$$k(d_i, d_j) = \varphi(d_i)\varphi(d_j)$$

The maximum entropy method searches for the conditional probability distribution of the class label c given a data instance that is as uniform as possible under given constraints. The probability distribution would be uniform if no limitations were applied. Each constraint shifts the distribution away from uniformity while bringing it closer to the data. Using features, constraints on the conditional distribution are determined from the training dataset. They have defined a feature fi(d, c) as a real-valued function of the training data instanced and the class label c. Similar to in MNB we can use token (AUD or AUD bigram) counts as features, where f_i is a function that equals zero if the token t does not appear in the utterance d and is equal to the number of token occurrences

$$f_i(d, c) = \begin{cases} N(d, t), & \text{if } t \in d \\ 0, & \text{otherwise} \end{cases}$$

Partha Pratim Barman et. al[3] proposed that next word prediction is a highly discussed topic in the current domain of Natural Language Processing research. Felix et. al (1999) used LSTM to solve tasks that were previously unsolvable by RNNs. They introduced forget gates to solve continual versions of these problems. Mikolov et. al(2010) presented a simple Recurrent Neural Network-based language model to improve the prediction of the next word in sequential data. In another work, Alex(2013) discussed the use of LSTM to generate complex long-range structured sequences. The author implemented the method for text and online handwriting prediction with an extension to handwriting synthesis. Sukhbaatar et. al (2015) introduced a Recurrent Neural Network to perform next word prediction on a text sequence.
Human beings don't ponder over new thoughts on their own every second. While going through a text, we acknowledge each word based on our understanding of the previous words. We have the capacity to relate and the thoughts are continuous. Traditional neural networks fall short in such scenarios, which is a massive disadvantage. Consider the following scenario: we want to categorize the type of event that occurs at each point in a novel. It's unclear how a typical neural network could use earlier plot events to inform future ones. Recurrent neural networks can be used to address this issue. For instance, let us examine a language model trying to predict the next word on the basis of the previous ones. If we are trying to predict the last word in 'The sunrises in the east, we don't require any other conditions as the east is the pretty obvious next word in this case. In these cases, where the gap between the relevant information and the place that it's needed is small, RNN can learn to use the past information.

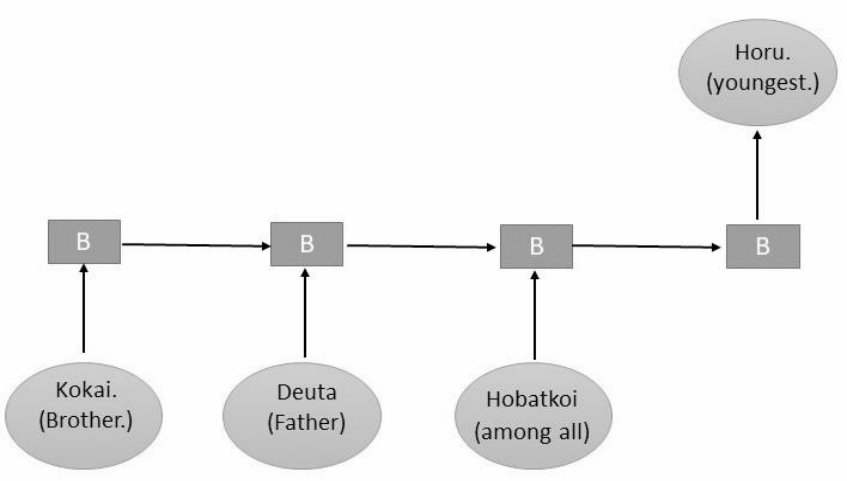

Fig An example scenario of the implementation.

Long Short-Term Memory Networks (LSTMNs) are a type of RNN that can learn long-term dependencies. They were introduced by Hochreiter & Schmidhuber(1997). All recurrent neural networks have the form of a chain of repeating

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 4, April 2022

DOI: 10.17148/IJARCCE.2022.11449

modules of a neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tank layer. LSTMs also have this chain-like structure, but the repeating module has a different structure. Instead of one neural network layer, there are four, each interacting in a unique way.
The cell state is the primary key of LSTM: it flows straight along the chain, with only slight linear interactions. It is relatively easy for information to just pass along unmodified in it. LSTM may add or delete information from the cell state using gates. They have a sigmoid neural net layer and a pointwise multiplication mechanism. LSTM has three of these gates, to protect and control the cell state. The first step in our LSTM is to decide what information was going to be thrown away from the cell state. This decision is made by a sigmoid layer called the forget gate layer. It examines ht1 and xt and returns a value between 0 and 1 for each number in the cell state Ct1. An output 1 means 'totally keep this,' whereas a 0 means 'absolutely get rid of this.' Here ht−1 is the output of the previous LSTM block, xt is the input and ht is the output of the current LSTM block, Ct−1 is the cell state output from the previous LSTM block, Wf is the weight vector and bf is the bias of forgetting gate layer. ft is the output of the forget gate layer.

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f)$$

Hemant Palivela proposed the following[4]

Paraphrasing can be subdivided into two tasks namely Paraphrase Identification and Paraphrase Generation. The Identification task is a discriminative type of task that determines whether a pair of sentences have the same meaning. In this task, the system may return a probability between 0 and 1, with a number closer to 1 indicating that the sentence pair is a paraphrase of each other and a value closer to 0 indicating that it is not. In some circumstances, the identification system generates a semantic score that, when normalized, can aid in sentence pair discrimination. Given a reference or input text, the Paragraph Generation job seeks to automatically produce one or more potential paraphrases. The goal is to develop paraphrases that are semantically similar and fluent.

The PI task is viewed as a supervised machine learning task and is modeled as follows:
Given a sentence pair ($S1$, $S2$), the aim is to find the target (1 or 0 which depicts the given sentence pair is a paraphrase of each other or not respectively) where the sentence $S1$ = { $w1$, $w2$, $w3$, …, wn} and $S2$ = { $w1$, $w2$, $w3$, …, wm }. It depicts that the length of the sentence may vary. A probability between 0 and 1 or a normalized semantic scoring mechanism can be used as the outcome.

The goal of the PI task is to create a candidate sentence from an input sentence. Given an input sentence or a reference sentence $S1$ where $S1$ = { $w1$, $w2$, $w3$, …, wn} , the aim is to generate one or more candidate sentences $S2$ = { $w1$, $w2$, $w3$, …, wm }, $S3$ = { $w1$, $w2$, $w3$, …, wo } , ... $S4$ = { $w1$, $w2$, $w3$, …, wp} . The length of the generated candidate sentences and the input or reference sentence may differ in this task as well.
The authors proposed a unified system architecture capable of performing both the paraphrasing tasks of identification and generation.

In the English language, the ParaNMT database has about 50 million sentential paraphrase pairs. A back-translation mechanism was employed to create the massive ParaNMT corpus. To extract texts written in Czech to English, a Neural Machine Translation (NLT) method was utilized. There are 404,290 sentence pairs in the Quora duplicate questions pair dataset. This data was divided into two sets: 70 percent training and 30% testing. The training data contains 283,003 sentence pairs, while the test data contains 121,287 sentence pairs. These lines were taken from two-year news clusters on the World Wide Web (WWW). Around 5800 sentence pairings are included in the final MSRP database. The training set has 4076 sentence pairs, whereas the test set contains 1725 sentence pairs. These three kinds of data are utilised to train the model.

By filtering and sampling the original data, the goal is to promote data diversity. The paraphrase generation model is seen to produce accurate paraphrases without repetition by maximising lexical, semantic, and syntactic variation in the training data. This allows the paraphrasing model to produce more different paraphrases with the same meaning but a larger vocabulary. To boost data diversity, the following changes are done to the training data. Remove sentence pairs that overlap by more than 60% unigram, bigram, or trigram. This reduces the likelihood of the final trained model copying the input sentence and increases the likelihood of creating varied paraphrases. Sentence-BERT was used to remove sentence pairs with very little semantic similarity (Reimers & Gurevych, 2020). This forces the final trained model to generate sentences that are semantically similar.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 4, April 2022

DOI: 10.17148/IJARCCE.2022.11449

Zejian Shi et. al[5] proposed that the full name of LSTM is Long Short Term Memory, which can selectively store and discard the information in the hidden layer of neural networks. In another word, it can decide the information storage time in neurons. The emergence of LSTM figured out the gradient appearing problem well and promoted the development of Recurrent Neural networks.

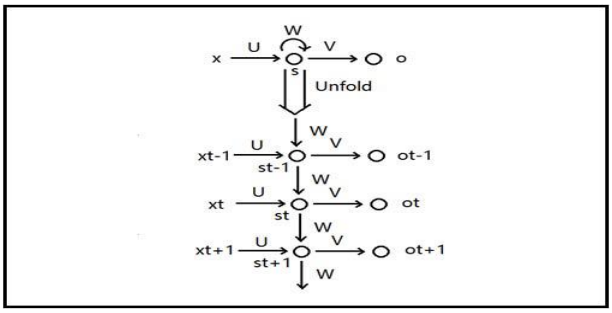

Fig. 1. The expansion graph of RNN

It is the process by which a conventional recurrent neural network develops into a network, as seen in Fig. 1. There are three layers in it: an input layer, an output layer, and a concealed layer. The "x" means the input layer. The "s" means the hidden layer and the "o" means the output layer. The hidden layer is calculated by the current input layer and previously hidden layer. As a result, we can deduce that the cyclicity of RNN is the key to understanding this graph. The formula shown in the graph is as follows. The function "f" in this expression, like the "tanh" function, is nonlinear.

$$s_t = f(U \cdot x_t + W \cdot s_{t-1})$$

$$o_t = g(V \cdot s_t)$$

LSTM is a particular recurrent neural network. Because the repeating module of a conventional RNN has a simple structure, LSTM replaces it with a more complex structure. Then let's compare the structure of LSTM with the structure of standard RNN. The explanation of LSTM combines my understanding with an article named "Understanding LSTM Networks".

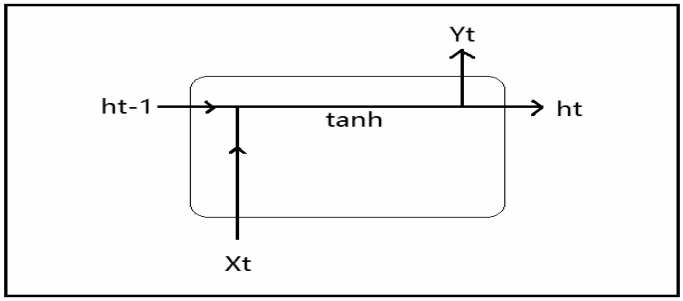

Fig. 2. The construction of standard RNN

This is the repeating structure inside the standard RNN. It is very easy to understand. Then I will introduce the structure inside the LSTM in four parts.

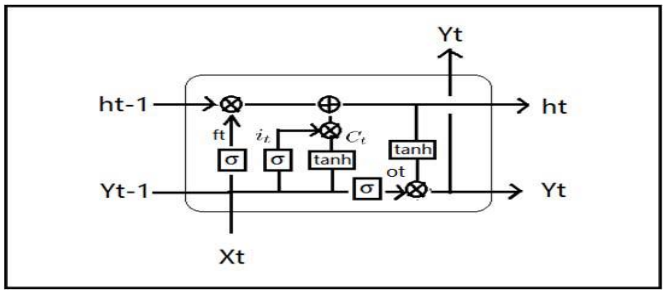

Fig. 3. The construction of LSTM

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 4, April 2022

DOI: 10.17148/IJARCCE.2022.11449

Mike Krey[6]stated that non-research sources could provide useful insights into connected concerns and future trends by thoroughly researching the fast-paced pace of IT GRC and its impact beyond academics. For this reason, the review focuses on literature grounded in science and practice. To classify existing approaches and identify their relevance to the identified problem, the taxonomy framework by Cooper [7] was applied. As a result, the review is conducted using different criteria. In the first stage, literature sources that directly connect to the health care sector and focus on integrated GRC are investigated to provide transparency in the review of the outcomes. This configuration contributes directly to the derived research problem.

The authors seek to empirically verify the gained insights from the previous areas, outlining the hospital as a complex and sensitive environment for the adoption of IT GRC. With a focus on the demand for a hospital-specific approach to adopting GCC, the implementation of IT GRC principles in the Swiss hospital environment. Although the separate elements of IT GRC are widely accepted, applied, and empirically validated in several industries, resulting in a positive impact on the enterprises' effectiveness, integrated IT GRC is still less widespread in the Swiss hospital environment. This investigation is associated with a survey conducted in 2009 by Krey and his colleagues [5] and allowed, therefore, concluding the progress of IT GRC management in Swiss hospitals over the last five years.
The results have been presented in the same format as the interview standards. To begin with, the findings revealed that IT GRC in health care is still all too often seen as the realm and sole responsibility of the CIO and the IT department. The results proved that IT GRC has not been utilized sufficiently by the executive management of many hospitals, especially the public ones. Thirty percent of hospitals (n=5) surveyed believed that only one-fifth of their business managers could explain their IT GRC arrangements.

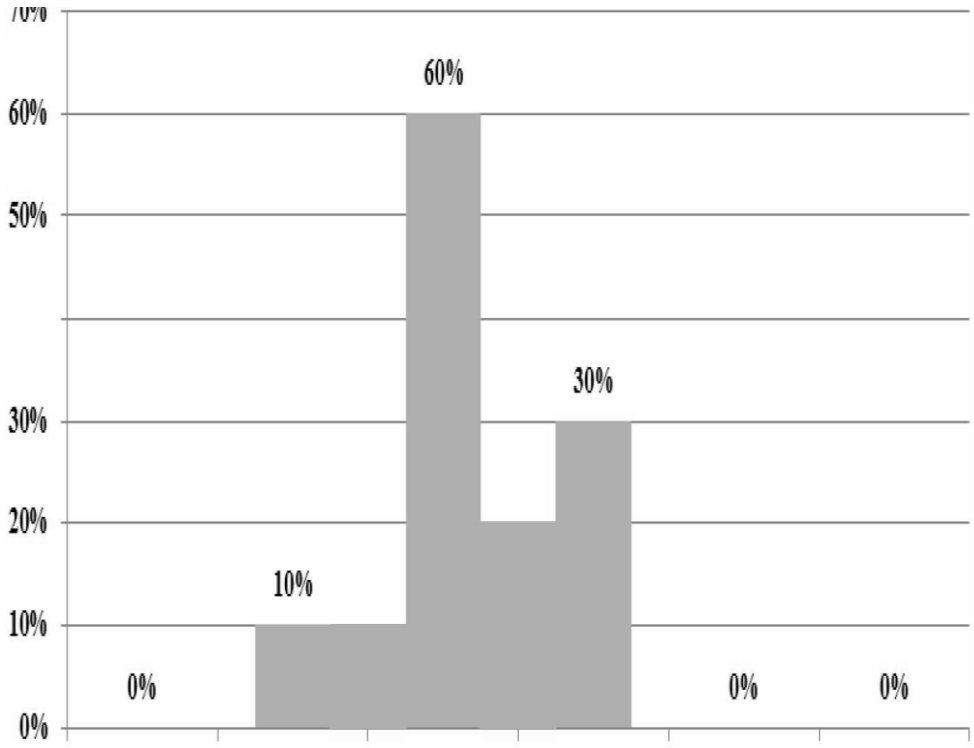

Figure 1. Classification of ITlL approaches

Consequently, related management questions, such as "What is the quantified risk potential and the related harm in the case that the HIS is unavailable?" can be answered. With the help of downtime estimation, an approximation of the longest acceptable downtime that the business (e.g., radiology department) can endure while remaining viable is assessed. Gained insights from active management of risk potentials consequently lead to decisions on strategic issues, such as the building of a new data center or endeavors towards outsourcing of IT capabilities. In this respect, IT governance provides roles and responsibilities, such as Chief Risk Officer (CRO) along with a dedicated security expert (e.g. Chief Security Officer, CSO) to ensure that IT risk management concerns are sufficiently addressed.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 7.39 ⁜ Vol. 11, Issue 4, April 2022

DOI: 10.17148/IJARCCE.2022.11449

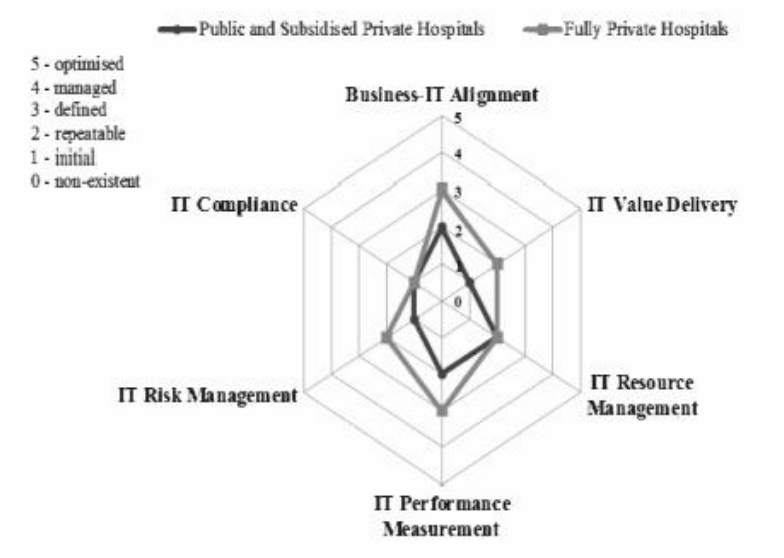

Figure 2. Comparison of IT GRC capabilities

III. FUTURE SCOPE

In the future, we'll use machine learning techniques to teach the algorithm, which should improve the outcomes. With supervised learning, conditional random fields can also deliver improved outcomes in the keywords extraction technique. Both of these ways will be used in the future to improve the current development. We also intend to create larger datasets than those used in normal life, test our models, and analyze the outcomes. Although the Assamese language is the focus of our research, the LSTM model can be extended to other indigenous languages as well. The TensorFlow platform is used to run the application, demonstrating RNN's capacity to perform sequential processing. If the program uses RNN to create a word-level language model, the later work will be easier.

IV. CONCLUSION

Next-word prediction utilizing Latent Semantic Analysis, as well as other fields of research, is fascinating. Some of the challenges can be overcome by employing conditional generation-based neural networks for paraphrase generation, and the goal of identification can be approached as a sentence-pair binary classification task. Because the suggested technique takes use of multi-GPU training and simultaneous assessments, the end-to-end system may be employed in a real-time production setting. LSTM is used to anticipate the next word in the phonetically transcripted Assamese language. This is being offered in order to examine and explore time management in electronic communication. Because it alleviates the computing cost and learning time, the suggested method can be deemed superior to the neural approach. The experiment determines the value of the learning rate, which is a hyper-parameter.

Finally, while complex and application-dependent, properly configuring the learning rate can lower the neural network's loitering time.

V. REFERENCES

[1] H. M. Mahedi Hasan et al. proposed **"**A Novel Approach to Extract Important Keywords from Documents Applying Latent Semantic Analysis**"** IEEE, 2018

[2] Vladimir Despotovic et al. proposed "Machine learning techniques for semantic analysis of dysarthric speech: An experimental study" Elsevier, 2018

[3] Partha Pratim Barman et al. proposed "An RNN based Approach for next word prediction in Assamese Phonetic Transcription" Elsevier, 2018

[4] Hemant Palivela et al. proposed "Optimization of paraphrase generation and identification using language models in natural language processing" Elsevier, 2021

[5] Zejian Shi proposed "The Prediction Of Character-Based Neural Network Language Model" IEEE, 2017.

[6] Mike Krey proposed "Next Word Prediction for Phonetic Typing by Grouping Language Model" IEEE, 2016.