Quest Journals
Journal of Software Engineering and Simulation
Volume 9 ~ Issue 4 (2023) pp: 76-82
ISSN(Online) :2321-3795 ISSN (Print):2321-3809
www.questjournals.org

Research Paper

QnA Framework Using Machine Learning

Anuj Negi[1], Arpit Yadav[1], Manas Saxena[1], Sudheer Singh[1],
Mr. Lucknesh Kumar[2]

[1]UG Students, Department of Information Technology, Ajay Kumar Garg Engineering College, Ghaziabad, Uttar Pradesh, India
[2]Assistant Professor, Department of Information Technology, Ajay Kumar Garg Engineering College, Ghaziabad, Uttar Pradesh, India

ABSTRACT

People have been attempting to obtain data from a sizable text database for a very long time. Transform data into the information we require. In the present search engines, when we search for anything, instead of providing the precise answer, it extracts keywords from our search and provides us papers or web pages linked to those phrases. Yet, because what we want is the exact answer, why does the user have to search for it? In other words, search engines focus more on retrieving entire documents. Nonetheless, a user frequently seeks an exact or detailed response to their query. For instance, instead of having to go through numerous online sites including the words "Holi," "festival," "year," etc. in order to discover the answer to the inquiry "When is Holi festival this year?," he would want to receive the response "March 9, 2022." That is, rather than the present method of document retrieval, a user wants information retrieval. When the answers to a query can be found in documents stored in a sizable text database, we take care of addressing those inquiries. We employ a machine learning approach to provide answers. A classifier that has been trained on a set of question-answer pairings is specifically used to categorize and rate response candidates.

Keywords—Data, Database, Machine learning, Natural Language Processing, Natural Language Understanding.*

***R**eceived 03 Apr., 2023; **R**evised 18 Apr., 2023; **A**ccepted 21 Apr., 2023 © The author(s) 2023.*
P**ublished with open access at **www.questjournals.org

I. INTRODUCTION

A Question Answering (QA) system is a piece of technology that, as opposed to providing a list of responses, offers the precise, succinct response to a question. In this setting, QA systems evaluate text similarity and respond to inquiries that are repeatedly posed in natural language. If we want to create a QA system that can use NLP and enhance machine learning algorithms to respond to specific or general inquiries, we can do this with the use of natural language processing technologies, such as Google's Bidirectional Encoder Representations from Transformers (BERT). We have a lot of question-answering systems today thanks to technological advancement, but they are insufficient. To spare us the hassle of digging up answers, we require a question-answer system that is accurate, exact, light-weight, and quick. It can be applied anywhere, including in robotics and search engines.

When we wish to look for information on a website, we use search engines to sift through relevant documents. In any case, since they display articles to us, we must examine those materials to see if they include the data we require. It's a difficulty. In order to find information accurately, the majority of search engines offer a question-answering feature [10]. The traditional area of research in information retrieval is finding the solution we're looking for in a sizable text database. When given a list of keywords, modern search engines return a large number of web pages that contain those keywords, leaving it up to the user to sift through the large number of sites and pages they return to get the information they need. Search engines focus more on retrieving entire documents. In reading comprehension, question-answering is a topic that is covered in another body of research. Long chapters can need a lot of reading and effort just to look up a few answers.

II. LITERATURE REVIEW

Recently, the two-stage retriever-reader QA methodology described by has gained widespread acceptance. (D.Chen et al., 2017). The reader component uses the information from the documents that the retriever section has pulled from a big database to offer the answer to the inquiry. Other frameworks exist as well, such generator and retriever-generator. In line with the development of pre-trained models (K.Guu et al., 2020; K.Lee et al., 2019), the work on two-stage Question Answer model structures aims to improve the execution of these designs using a variety of methodologies. BERT has been attracting interactive attention. (M.Gardner et al., 2019; W.Yang et al., 2019). Pre-trained models have been used in QA systems, including Bidirectional En-coder Representation from Transformers (BERT), DistilBERT, and RoBERTa. (M.Zaib et al., 2020). Machine learning has expanded in many application domains, and it is now widely employed in all stages of QA systems. (Z. Huang et al., 2020).Numerous natural language problems have significantly improved thanks to the pre-trained model of natural language processing. (X.Qiuet al., 2020). Using MLM and NSP, BERT is able to obtain a vector representation of the questions.(Jacob Devlin et al., 2019).

III. PROPOSED METHODOLOGY

A. Module Description

1. User sign-up: The user sign-up is for new users to create account enter details their name email-id and a new password so that they can login with same credentials afterwards.
2. User login: The user login feature allows users who have already registered for the app to log in at the last place they were logged out by simply entering their email address and password.
3. Add a passage or article: Users may add a passage or article to the space provided from which they wish to pose inquiries.
4. Ask question: In order to avoid reading a lengthy essay for few replies, the user can compose a question and ask it from the passage.
5. Get answer: Through bert, the user can obtain accurate replies to their queries from the backend.

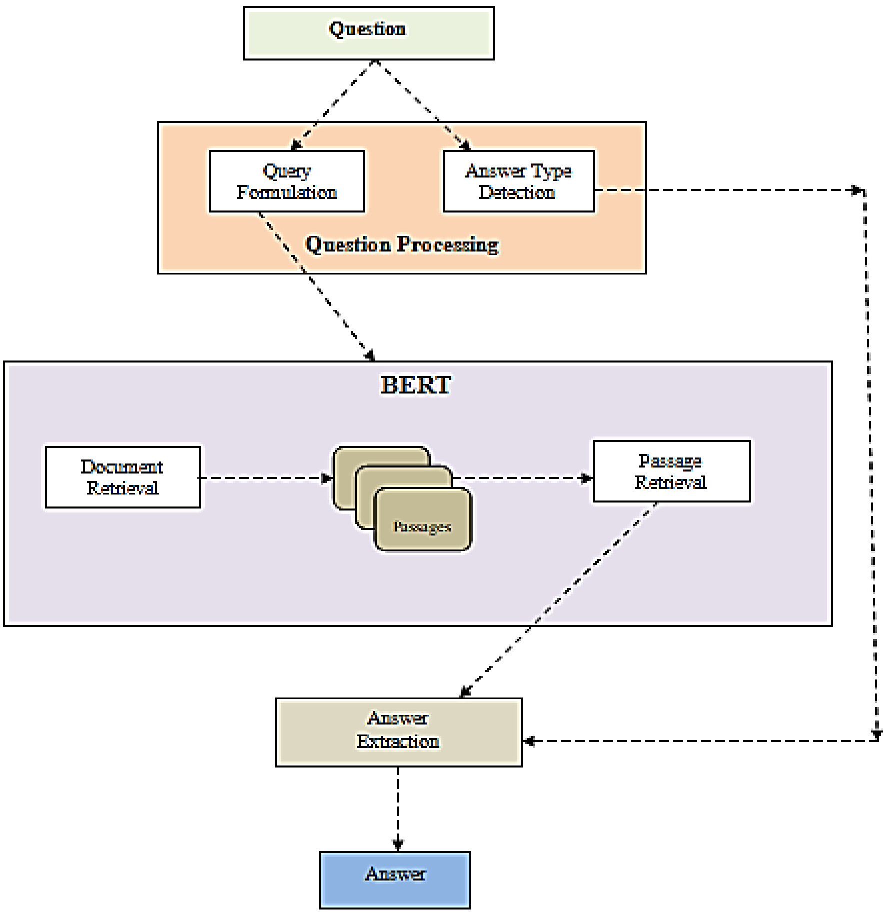

Figure1: *Block Diagram*

IMPLEMENTATION

Models Analysis

HuggingFace's Simple Transformer library was used in our work. We can train and evaluate transformer models using straightforward transformers. Three models—DistilBERT, BERT, and ROBERTA—were examined.

1. *DistilBERT*

The blog post restricted, speedier, and economical introduced the DistilBERT model as an improved BERT variant. A Transformer model created by distilling BERT base is called DistilBERT. Bert-base-uncased performs 60% more quickly than DistilBERT while keeping over 95% of BERT's performance as measured by the GLUE language understanding benchmark [3]. Bert-base-uncased has 40% more constraints than DistilBERT.

2. *BERT*

BERT, or Bidirectional Encoder Representations from Transformers, is a machine learning method for analysing natural language that Google offers. It aids computer systems or machines in comprehending simple human language, to put it simply. When technology advances, it assists in identifying patterns in those sentences that are necessary for machines to comprehend humans and converse [2].

3. *RoBERTa*

Robustly Optimized BERT Pre-training Method is known as RoBERTa. Facebook created and maintains it in order to speed up the data supply to the BERT engine. In contrast to the BERT approach, RoBERTa uses natural verses rather than human-labeled data. This extra capability makes it quicker to train on large amounts of data. [7]

After training these models on our data we got the result displayed in fig 4.

These findings in Fig. 4 show that bert-based-cased obtained the most right and least incorrect results with the least evaluation loss. Hence, in our application, we employed the BERT model.

A. Word2vec based Question Answer Model

Word2Vec uses a neural network to extract word associations from a large corpus of text. After training, this model can recognise similarities between words, predict synonyms, and determine the cosine correlation between various words. The system can recognise the meaning of the language thanks to the ability to calculate word similarity. This might be used to create a question-and-answer format. With our dataset, we were able to create word2vec embeddings for a question-answering design:

The input data that has been turned into an inventory of lists is first subjected to fundamental data manipulation in Python. This piece of data is given into the Word2vec model. The embedding size is kept constant at 100 and the context window is kept at a size of 8. After that, the model is trained for 50 epochs. Then we start responding to queries. Our question was cleaned up, then we submitted it to word2vec as single words. An embedding for the question was produced by adding and calculating the average of the formed embeddings. Next we moved on to the parallel passage and treated it in the same way that we treated the question. Then, we employ the same method to find embeddings for each sentence in our data. Once we have the embeddings for the question and each line in the answer text, we use cosine similarity to determine the connection between the embeddings of the question and each passage phrase. The output of the model for the given question is the phrase that has the highest correlation with the query.

This technique identifies the passage's sentence that most closely resembles a given question in a Word2Vec embedding and outputs it as the response in the Word2Vec model. It is a straightforward and moderately effective model.

B. Technologies used

1. ReactJS

(Frontend) A JavaScript library called ReactJS is mostly concerned with user interface. It is beneficial to design lightweight, readable, and understandable code that is reusable. It encourages the development of reusable UI elements that provide data that changes over time. React is known as the MVC's "V". React is the way to go if you want a simpler programming model and greater speed. React enables the creation of both mobile and web applications with React Native. Instead of using traditional data binding, React uses one-way reactive data flow, which reduces boilerplate and is simpler.

2. Firebase:

For our app, static and dynamic scope, and microservices, Firebase Hosting offers secure and quick hosting. For developers, Firebase Hosting is a product as a service (PaaS) web host.

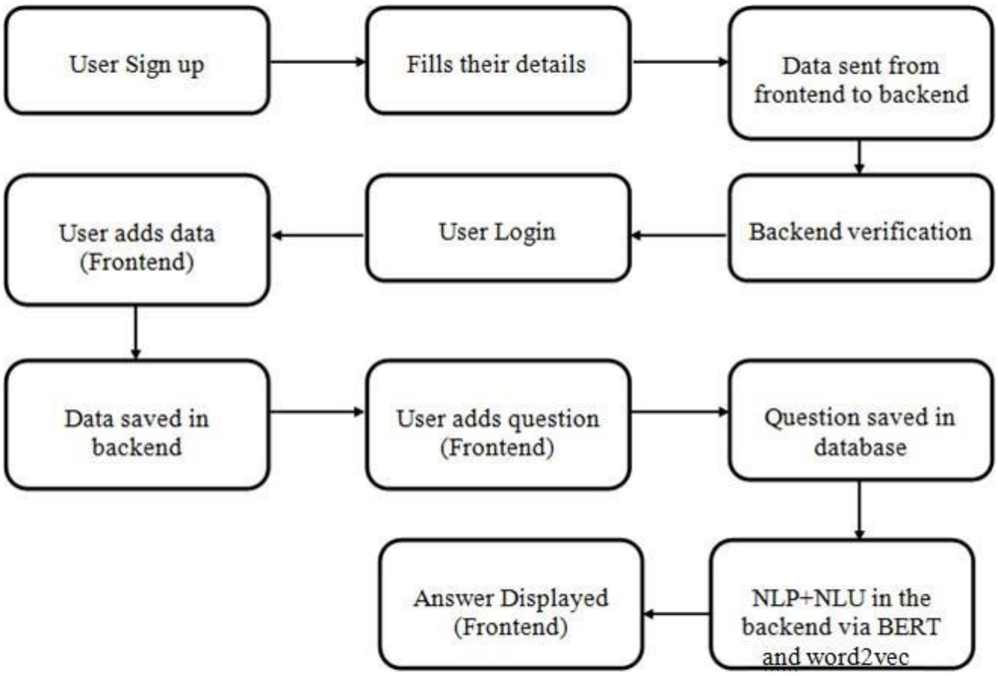

Figure2: *Architectural diagram of web-application*

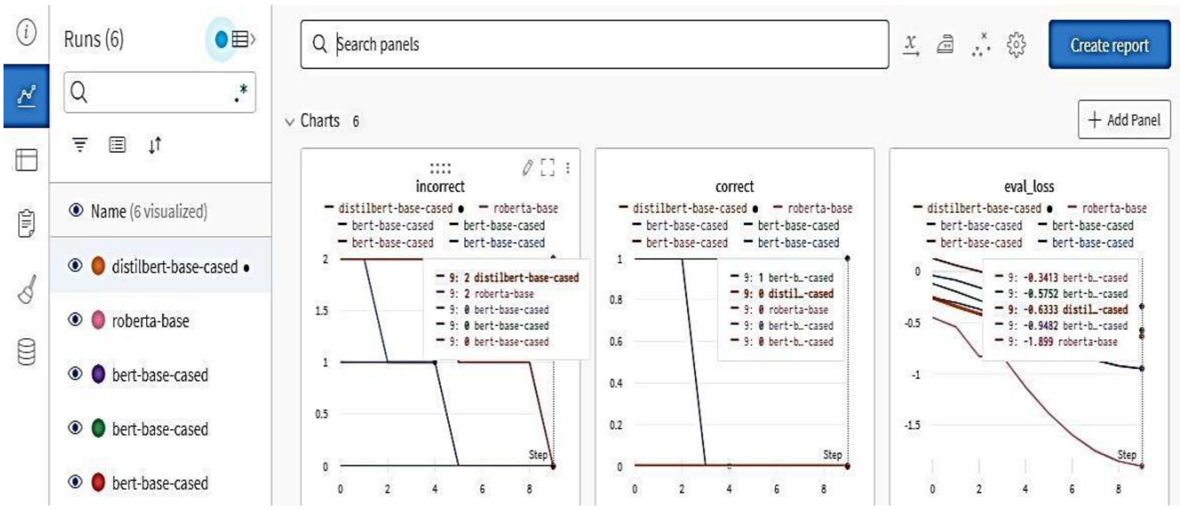

Figure 3: *Model analysis*

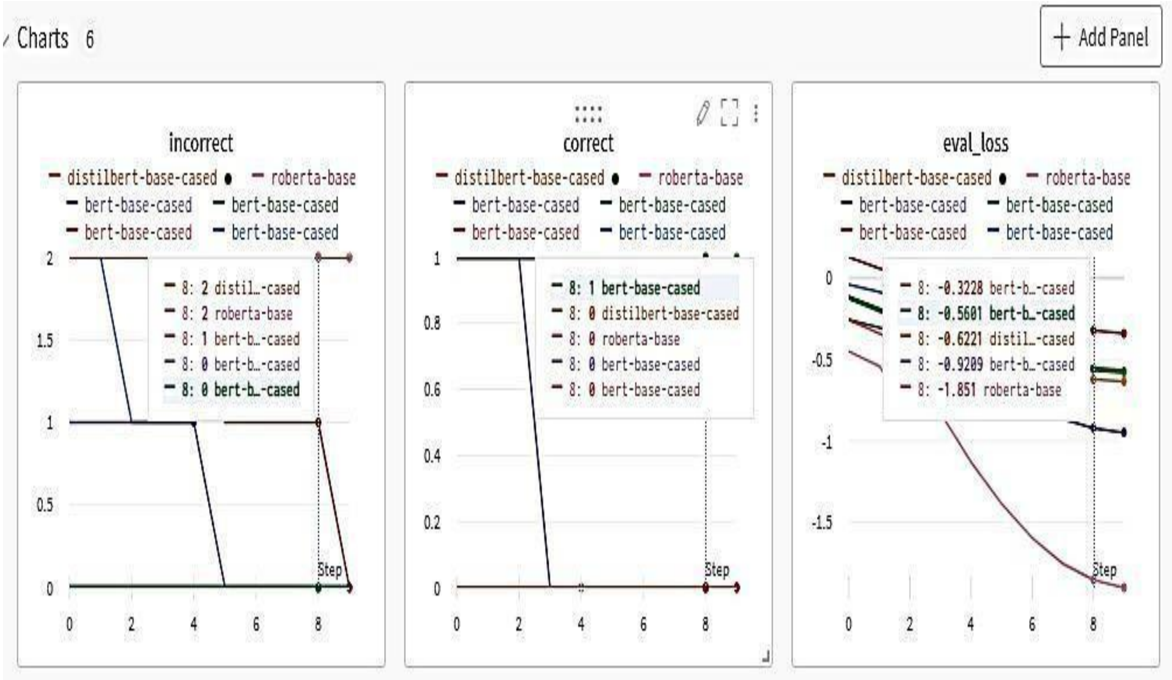

Figure 4: *Result of model analysis*

To set up and host microservices on Firebase, we may also use Cloud Functions or Cloud Run in conjunction with Firebase Hosting. We can instantly deploy web applications and supply both static and dynamic composition to a global CDN with a single command. (content delivery network).

3. NodeJS

One of the most well-known open-source online development environments, NodeJS is utilised on both the front and back ends of websites. We now have online applications with real-time updates, which greatly simplifies the process of creating apps. In two-way communication, the client and server can connect to one another and exchange data without any difficulty.

4. BERT

Bidirectional Encoder Representations from Transformers, or BERT, is a machine learning method for natural language processing that Google offers. We selected this model because it produced the most accurate results when trained by us [2].

5. Word2vec

Word2Vec mines a big text corpus for word associations using a neural network. After some training, this model is capable of identifying word similarities, forecasting synonyms, and calculating the cosine correlation between various words. The system's capacity to determine word similarity allows it to recognise the semantics of the language.

C. Algorithms used:

1. Natural Language Processing (NLP)

Human speech is broken up into pieces in natural language processing so that the sentence structure and decrypting words can be interpreted and comprehended in relation to one another. This enables computers to comprehend and interpret spoken or written language similarly to humans [5]. Before NLP technologies can generate a sense of human languages, data scientists must conduct a few powerful NLP activities, such as stemming, lemmatization, and tokenization. Skip-gram and CBOW are two NLP algorithms that we used in our model.(Continuous Bag Of Words).

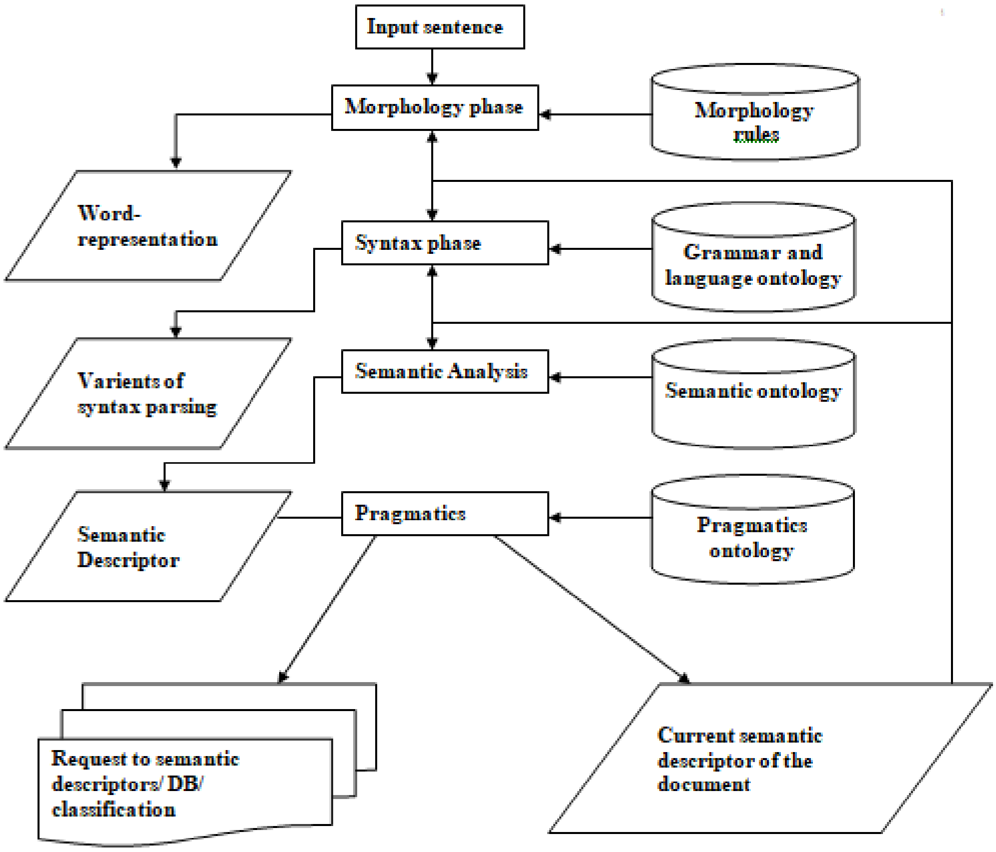

Figure 5: *General scheme of natural language processing*

2. Natural Language Understanding (NLU):

Software for natural language understanding computes input into sentences from text or speech. It makes it easier for computers to understand popular languages without the pre-established structure of computer programs, such as English, Hindi, and Italian. NLU helps people and computers communicate. Additionally, NLU enables computers to communicate with people in their own languages [5].

IV. RESULT

In the context of skimming through lengthy texts, question-answering emerges. It might be time-consuming and labour-intensive to read through extensive documents in pursuit of a few answers. A tonne of time and effort will be saved by this app. Huge amounts of data will be converted into the information we need. By converting facts into information, the passage will provide the answers to our query. Finding the information we need in a large database is a hassle. Thus, having an automated system for answering questions that allows us to precisely discover information that is pertinent to us makes daily tasks easier. Word2Vec preserves the semantic significance of various words inside a document. Information about the context is retained. The fact that the embedding vector's size is so short is another important advantage of the Word2Vec approach [6]. Contrarily, BERT will produce two distinct vectors for the word bank utilised in the two different situations. By combining the strengths of the two models, word2vec and BERT, we may produce a model for data retrieval that is more exact.This strategy produced superior outcomes than utilising either of the two models, as evidenced by the application model.

V. CONCLUSION

This machine learning-enabled responsive application will primarily give users the kind of specific, quick response or information they desired, saving them a lot of time and simplifying the task of information retrieval. This could also help to respond accurately, in search engines to find the information needed by the user, as well as automated robots can reply in a much more decisive manner, which is quite potent.

REFERENCES

[1]. Jacob Devlin et al. (2018). BERT: Pre- training of deep bidirectional transformers for language understanding, Computation and Language, 1, Available at: https://arxiv.org/abs/1810.04805?msclkid=e 2a628f1b63311ecb252cc1cdf4a6e1d.

[2]. B. BarlaCambazoglu et al. (2020). A review of public datasets in question answering research, ACM SIGIR Forum, 54(2), 1-23, Available at: https://dl.acm.org/doi/abs/10.1145/3483382.3483389?msclkid=6a3f37aab63711ec9331 b067a5ffe708.

[3]. Waheed Ahmed Abroa et al.(2022). Natural language understanding for argumentative dialogue systems in the opinion building domain, Knowledge-Based Systems, 242, Available at: https://www.sciencedirect.com/science/artic le/abs/pii/S0950705122001149?msclkid=85 b071bfb63511ecb5820fd6d0401f6c

[4]. Alex Wang et al. (2018). GLUE: A multi- task benchmark and analysis platform for natural language understanding, Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, 353-355, Available at: https://aclanthology.org/W18- 5446/?msclkid=c8eb734bb63411ec997e3d5 bfe2f46d6.

[5]. RishavChakravarti et al. (2020).Towards building a robust Industry-scale question answering system, Proceedings of the 28th International Conference on ComputationalLinguistics: Industry Track, 90- 101,Available at :https://aclanthology.org/2020.coling- industry.9/?msclkid=fe681145b63711ec9f7 ebc20eb6ae226.

[6]. Dhruvil Karani (2018), "Introduction to Word Embedding and Word2Vec", [Online] Available at: https://towardsdatascience.com/introduction-to-word-embedding-and-word2vec- 652d0c2060fa.

[7]. "RoBERTa: An optimized method for pretraining self-supervised

[8]. Chrissy Kidd & Bharat Saxena(2021), " Machine Learning & Big Data Blog - NLP vs NLU: What's The Difference? " [Online] Available at: https://www.bmc.com/blogs/nlu-vs-nlp- natural-language-understanding-processing/?msclkid=9ce6edc4b63811ec8e 97ee7196a3dea7.

[9]. "Building an application of question answering system from scratch" [Online] Available at: https://towardsdatascience.com/building- an-application-of-question-answering- system-from-scratch-2dfc53f760aa.

[10]. Yana Arbuzova (2020), "3 Winning Factors for Successful Development of Startups" [Online] Available at: https://medium.com/sigma-software/3- winning-factors-for-successful- development-of-startups-6c0f2e0cb120.

[11]. NLP systems",[Online] Available at: https://ai.facebook.com/blog/roberta-an- optimized-method-for-pretraining-self-supervised-nlp-systems/.