A Research Project

on

The Mathematical and Statistical Procedures for

Bengali Informative Intelligence Bot

Course Code: A.MTH-4205

Shanto Sutra Dhar

Roll No: ASH1506039M

Session: 2014-1015

Year of Examination 2019

Bachelor of Science(Honors)
in
Applied mathematics
Department of Applied Mathematics
Noakhali Science & Technology University

- Dedicated to my parents

CERTIFICATION

This is to certify that the research work entitled **"The Mathematical and Statistical Procedures for Bengali Informative Intelligence Bot"** is an original research work carried out by **Shanto Sutra Dhar (ASH1506039M)** the author under my supervision in partial fulfillment of the requirements for the degree of Bachelor of Science (Honors) in Applied mathematics under the Department of Applied Mathematics, Noakhali Science & Technology University.

Supervisor

Md. Jashim Uddin

Assistant Professor

Department of Applied Mathematics

Noakhali Science & Technology University

Noakhali - 3814

Declaration

I, Shanto Sutra Dhar (ASH1506039M), hereby declare that the project work carried out by me and submitted to the Department of Applied Mathematics, Noakhali Science and Technology University, Sonapur-3814, Bangladesh for the partial fulfillment of the requirements for the degree of Bachelor of Science (Honors) is an original work except where due reference is made and has not been submitted to any University/Institution for the award of the degree.

Shanto Sutra Dhar

Date: ……………………

ACKNOWLEDGEMENT

At first I express my deepest sense of gratitude to almighty Allah for giving me the strength and composure to finish the task within the schedule time. It would be a blunder not to mention those names I have met as the spirit of this project completion.

I express heartfelt gratitude to my honorable teacher and project supervisor Md. Jashim Uddin, Assistant Professor of Department of Applied mathematics, Noakhali Science & Technology University for his tireless supervision, guidance and inspiration throughout the entire study period. I am also grateful to him for his boundless patience in preparing and reviewing the project.

I am thankful to the other teachers and the stuffs of my department for their help and support. I also want to thank my parents and siblings for their encouragement, support and attention.

Sometimes I got myself out of track and clock beat me a lot for the completion of the project. Hence I'm also thankful to all my project partners who aided me in any case in doing work in time.

In case of any unwilling and erroneous presentation, I confess and promise to work more sincerely in future.

June, 2019 **- Author**

ABSTRACT

The Bengali Informative Intelligence Bot (BIIB) is an effective Machine Learning (ML) technique that helps a user to trace relevant information by Bengali Natural Language Processing (BNLP). In this research paper, we introduce two mathematical and statistical procedures for BIIB based on information of Noakhali Science and Technology University (NSTU) that is significant mathematically and statistically. In the preprocessing part, this research paper is demonstrated by two algorithms for finding out the lemmatization of Bengali words such as Trie and Dictionary Based Search by Removing Affix (DBSRA) as well as compared with Edit Distance for the exact lemmatization. We present the Bengali Anaphora Resolution system using the Hobbs' algorithm to get the correct expression of consequence questions. In order to reduce the time complexity of searching questions and reply from inserted information, we have used Non-negative Matrix Factorization (NMF) as the topic modeling technique, and the Singular Value Decomposition (SVD) as to reduce the dimension of questions. TF-IDF (Term Frequency-Inverse Document Frequency) has been used to convert character and/or string terms into numerical values, and to find their sentiments. For the action of Chatbot in replying questions, we have applied the TF-IDF, cosine similarity and Jaccard similarity to find out the accurate answer from the documents. In this study, we introduce a Bengali Language Toolkit (BLTK) and Bengali Language Expression (BRE) that make the easiest implementation of our task. We have also developed Bengali root word's corpus, synonym word's corpus, stop word's corpus, and collected 74 topic related questions and answers from the information of NSTU which are actually our inserted informative questions. For verifying our proposed systems, we have created 2852 questions from the introduced topics. We have got 96.22% accurate answer by using cosine similarity and 84.64% by Jaccard similarity in our proposed BIIB.

CONTENTS

Chapter One

Introduction

Objective

- To explain the importance of BNLP
- To access the knowledge of Chatbots
- To summarize the whole theme briefly

INTRODUCTION

A chatbot is an artificial intelligent (AI) system that can interact or “chat” with a human user in natural language such as English [1]. The first chatbot ELIZA was designed by Joseph Weizenbaum in 1966 [2]. It was originally created to simulate a psychotherapist [3]. After ELIZA, different chatbots were created such as A.L.I.C.E. and Mitsuku. Most of those chatbots were developed by using Artificial Intelligence, which is a programming language that enables chatbots to recognize patterns in the input sentences, and respond with sentences from a template.

Chatbots are becoming popular and they are now creeping into our smartphones. People spend a lot of time on the apps installed in the smartphones everyday. In China, senior users of mobile phones may not know how to use every apps, however, most of them use the chatting app like WeChat frequently [4]. Even though they may not know how to type text messages on the mobile phones, they can use voice chat and images to express their ideas.

Chatbots are also used in many types of applications. They include applications for customer service in ecommerce websites, museum guides, language learning or chatting for entertainment purpose. Digital Assistants driven by Artificial Intelligence (AI) are becoming increasingly popular—Siri (Apple, 2011), Cortana (Microsoft, 2015), Google Now (2012), Alexa, (Amazon 2015) are among the top voice-enabled digital assistants. These assistants can send users updates on the weather, help them know about the traffic situation on the way from/to work or home,

Digital assistants are changing the way people search for information, making it part of regular conversation. These existing assistants however are mostly productivity oriented, designed to support users in completing a range of tasks. Chatbots have also become much more intelligent. They can understand human users very well and they can provide human-like responses. This is due to the rapid development of artificial intelligence and other related technologies. Many investors understood the potential of AI and they have made significant investments to harness the technology.

In Bengali language processing, the Bengali Intelligence Bot has been formed by three main modules: Informative questions and user questions processing, topic modeling and dimension reduction, revealing the TF-IDF model and regulating the best reply of the question by using the cosine similarity.

For example:

User Question-1**:** বাংলাদেশের সবচেয়ে বড় ছাত্রী হল কোথায় অবস্থিত?

BIIB Answer-1: নোবিপ্রবিতে অবস্থিত ।

User Question-2: এটির অডিটোরিয়ামের নাম কি?

BIIB Answer-2: বীর মুক্তিযোদ্ধা হাজী মোহাম্মদ ইদ্রিস অডিটোরিয়াম।

Different techniques are held for the sake of pre-processing of Bengali Natural Language, e.g., Anaphora, Cleaning, Stop Words Removing, Verb Processing, Lemmatization and Synonyms Word's Processing. In order to obtain a perfect Anaphora Resolution, the famous Hobbs' algorithm is imparted. In lemmatization action, we describe two procedures with the lowest time and space complexity. We have applied the topic modeling technique so that a question find its relevant topic and it does not require to search all questions but its related topic's questions. The topic modeling helps the machine reduce time complexity and answer a question at an instant time. To reduce the dimension of a question and information, we have used the SVD so that we can minimize the time and space complexity of a program. It also helps understand and calculation with simple way. The TF-IDF is used to notice the effect of words in documents and construct the perfect vector for cosine similarity. Cosine similarity comes from the concept of the dot product of vectors.

The contributions are summarized as follows:

- We have introduced mathematical and statistical procedures for BIIB based on Informative questions.
- We have used the NMF and SVD to reduce time and space complexity as well as instant answering of questions.
- For the preprocessing of data, we have applied Hobbs' algorithm, Edit Distance, Trie and DBSRA.
- We have introduced TF-IDF, cosine similarly and Jaccard similarity to find the best reply of a question.
- For the easiest BNLP, we have developed BLTK and BRE tools.
- For the easiest explanation, we take a unique example in every term of this paper.

Chapter Two

Related Work

Objective

- To discuss the allied works done before

RELATED WORK

Visual question answering is a QAS which includes in natural images, synthetic images, natural videos, synthetic videos and multimodal contexts [5]. WIKIQA is a Challenge Dataset for Open-Domain Question Answering system [6]. In Bio-medical QAS question types including factoid, list based, summary and yes/no type questions that generate both exact and well formed 'ideal' answers [7].

Zhou and Hovy [8] discuss a summarization system for technical chats and emails about Linux kernel that is used Internet Relay Chat and use clustering to model multiple sub-topics within a chat log.

Abu Shawar and Atwell [9] discuss the Artificial Linguistic Internet Computer Entity based chatbot called ALICE. It's English conversation patterns is stored in new file which named Artificial Intelligence Mark-up Language (AIML). The AIML is a derivative of Extensible Mark-up Language (XML) that enables people to input dialogue pattern knowledge into chatbots.

D. Aimless and S. Umatani [10] discribed the usage of AIML files that discussed the features and functionality of every files associated to build AIML based chatbot.

Thomas T. [11] provided a irregular responses chatbot which is planned in a manner that for single template.

Rashmi S and Kannan Balakrishnan [12] has illustrated a curious chatbot that finds the missing information and identification of missing data which provides same accurate response.

Md. Shahriare Satu and Md. Hasnat Parvez [13] introduced a review of integrated applications with AIML based chatbot.they used the applications AIML chatbot instead of human beings to interact with give solution of their problems and customers. They also said about the low cost and configuration of AIML based chatbots.

Wei Yun Gang,Sun Bo, Sun Ming Chen and Zhao Cui Yi [14] shows the Chinese Intelligent Chat Robot Xiao Hui-hui. The Chinese partitioning system and lack of dataset is the most difficult problem for them during this project.

Salvatore La Bua [15] has proposed an approach of LSA that applied its large amount of documents. It also describes the related words in the vector representation of corpus.

Hadeel Al-Zubaide and Ayman A. Issa [16] proposed an ontology based methodology that provides scalability and interoperability features for unlimited support of different domains.

Augello et al. [17] proposed a social chatbot model for a communicative skill learning game based on “Social practice” that can choose the most appropriate dialogue.

Hill J. et al. [18] shows a different chatbot such as human-chatbot conversations lack in content, quality and vocabulary and human-human conversations over IM.

Li Ka-Shing, the Hong Kong billionaire. He invested in several startups focusing on AI [19].

Jack Ma, the founder of Alibaba. He invested in an Israeli startup using AI to evolve ecommerce search technologies [20].

Dr. Kai-Fu Lee, a famous tech investor. He invested in several investments on AI startups that focus on the development of AI [21].

Pan, X. et al. [22] discussed a virtual agents chatbot. They suggested a projection based system virtual party like an environment. In this system, a conversational male user approached by a female character called Christine. Christine also can formulate personal questions and statements. This female character is interested in a male user that involves smiles, headnotes, eye contact, leaning towards.

Crutzen, R. et al. [23] studies an information chatbot that helps users to know about sex, drugs, and alcohol use among adolescents. This chatbot also called question related chatbot or search engines.

The best morphological analyzer for Russian, My stem, is based on Zalizniak grammatical dictionary [24]. This dictionary contains a detailed description of ca. 100,000 words that include their inflectional classes. My stem analyses unknown words by comparing them to the closest words in its lexicon. The ‘closeness’ is computed using the built-in suffix list [25]. A morphological analyzer of modern Irish used in New Corpus of Ireland is based on finite-state transducers and described in [26].

Nowadays QAS based on several method such as TF-IDF[27]which is a statistical method based on large scale corpus and Sementic Dependency[28],Deep learning method[29].

In chinesses language, Recurrent Neural Networks (RNNs) form an expressive model family for processing sequential data. it have been widely used in many tasks, including machine translation , image captioning , and document classification [30].

Chapter Three

Background Study

Objective

- To introduce all terms, methods, algorithms etc. used in this project

BACKGROUND STUDY

3.1 Bengali Natural Language Processing

Bengali (endonym Bangla) is the seventh most spoken language in the world. It is the most spoken as well as state language of Bangladesh, and also spoken in few parts of India like West Bengal, Tripura and Assam. But it is really a matter of great mournful that the Bengali language is deliberated as a low-density language because of the insufficient digitized text element in the Bengali language. However nowadays Bengali Natural Language Processing (BNLP) and AI are the most interesting research fields in Bangladesh [31]. Developing an ideal Chatbot needs a huge database of Bengali language. Because of lacking a large Bengali conversation corpus, a proper Chatbot has not yet been possible to invent. So the main goal of our task is to develop an ideal chatbot in Bengali language which will able to interact in Bangla language as well as building a corpus in Bangla. We name it "BIIB".

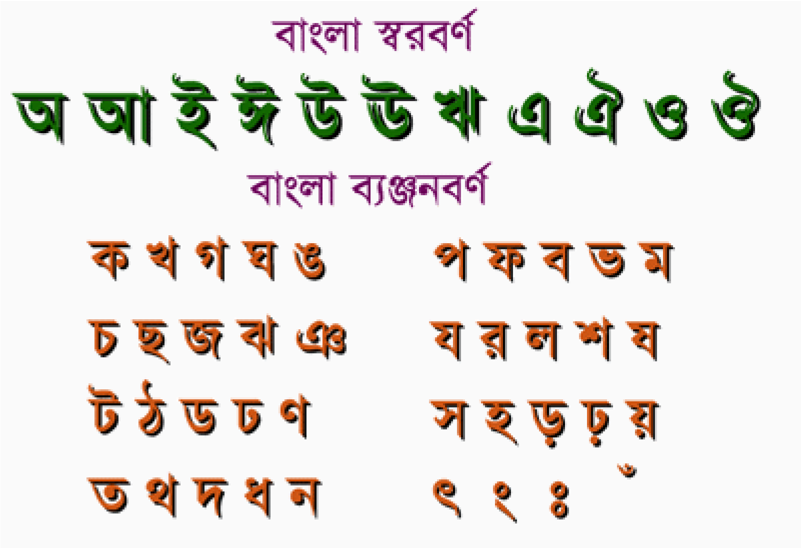


Fig. 1: Bengali Alphabet.

3.2 Lemmatization

Modern days are the days of automation with the help of computers. For automation, it is very important to develop computer programs to process and analyze large amounts of natural language data. However, this natural language processing (NLP) faces huge challenges as it involves good understanding of natural languages. Bangla is one of the complicated languages in the world. But there is no effective lemmatization technique in this language processing [32]. Lots of affixes (suffixes and prefixes) are there in a word in this natural language.

Lemmatization is a simplification process for finding out the extract root-word in natural language understanding. Lemmatization has been used in a variety of real world applications such as text mining, Chatbot, questions and answering etc. Usually lemmatization is done by dynamic programming based on Levenshtein distance [33] and data structure based on Trie algorithms [34].

In this project paper, we have used an effective lemmatization algorithm for the Bengali natural language processing. At first we have slightly modified the Trie algorithm based on prefixes. After that we proposed a mapping based new algorithm titled as Dictionary-Based Search by Removing Affix (DBSRA).

3.2.1 Levenshtein Distance

In computational linguistics and computer science, Levenshtein Distance or Edit Distance is a procedure where dynamic programming technique is used to measure the smallest figure of single-character that is enunciated to edit or change between two words (i.e. Insertions, Deletions or Substitutions). In the applications of NLP, Levenshtein Distance is widely applied for the correction of the spelling of a word [35]. There are two ways for the implementation of Levenshtein Distance as brute force and dynamic programming. Brute force way takes a long time complexity. For this case, in our propound work, dynamic programming is used to implement the Levenshtein Distance so that we might get a lower time complexity.

$$or, \qquad lev_{a,b}(i,j) = \begin{cases} max(i,j) & if\, min(i,j) = 0 \\ min \begin{cases} lev_{a,b}(i-1,j) + 1 \\ lev_{a,b}(i,j-1) + 1 \\ lev_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \end{cases} \qquad (1)$$

Equation (1) is called Levenshtein Distance between two strings.

Here ai and bj are indicators. If (ai == bj), it means the contained characters of ai and bj are equal and there is no need to change; so the edit = 0. Otherwise if (ai != bj), then the edit = 1. In Bengali lemmatization, we check the whole corpus of root words with the input sentences via the help of Levenshtein Distance. The minimum edit cannot be less than zero. If a corpus's word and an input word are equal then the edit will be zero (edit = 0) or otherwise 1.

3.2.2 Trie

Trie or Prefix Tree is a tree-based data structure, and exoteric for the mechanisms to store data. A trie stores words with a common prefix under the same sequence of edges in the tree, eliminating the same prefix each time for each word. In a trie, a data structure accommodates and retrieves all possible lemmas where every node contains a single character and every two nodes are connected with a single edge. Multiple branches maybe connected from every node. If we want to add a word to a trie, we first need to find the common prefix which is stored in this trie, and travel the whole common prefix. Hence for the new character a new branch of a node will be created. The 'End' in Boolean is always 'True' without the last character of a stored word in a trie

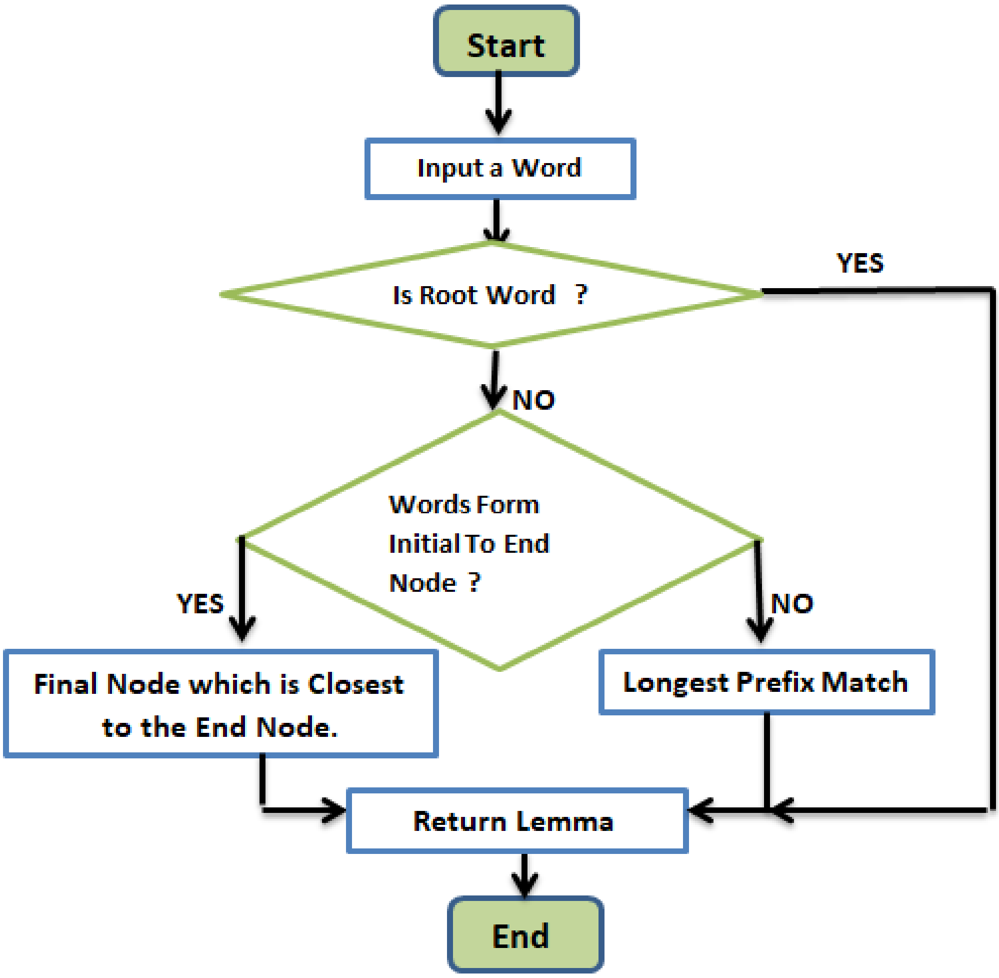

Start
Input a Word
Is Root Word ?
YES
NO
Words Form Initial To End Node ?
YES
NO
Final Node which is Closest to the End Node.
Longest Prefix Match
Return Lemma
End

Fig.2: Flow Chart of Trie.

3.2.3 Dictionary Based Search by Removing Affix

We fabricate a new lemmatization mapping algorithm that we call as Dictionary Based Search by Removing Affix (DBSRA).

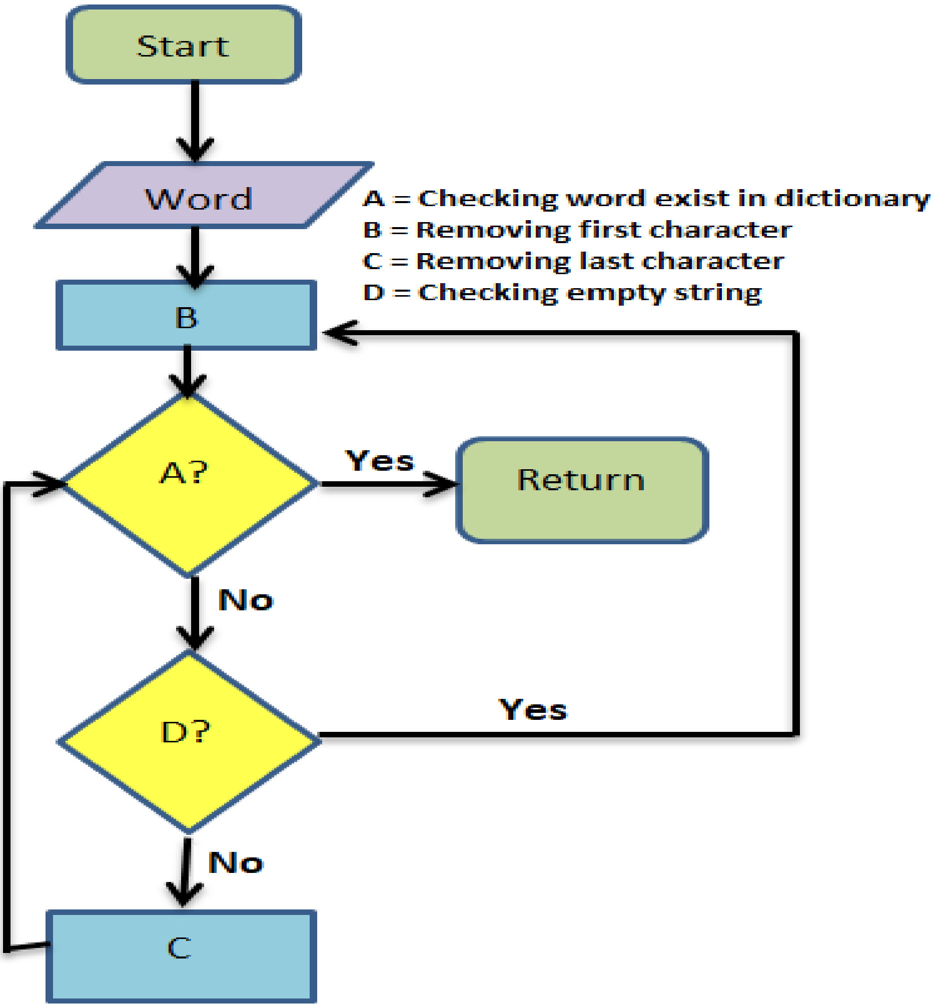

Start
Word
A = Checking word exist in dictionary
B = Removing first character
C = Removing last character
D = Checking empty string
B
A?
Yes
Return
No
D?
Yes
No
C

Fig. 3: Flow Chart of DBSRA.

In this technique, at first the i-th character is dismissed from the given word (which we want to transfer into root word); i = Length of word - 1, and remove j-th character; j = Length of word, e. g., n, n-1, n-2, ..., 1; (where i, j = 0, 1, 2,....). Then the algorithm investigates into our corpus whether the intentional word is in the root words corpus or not. But when the desired word is in massive volume, the searching will not be well performed and time complexity will be high. So the whole corpus has been mapped by the assist of dictionary-based program. As a result of mapping, the word can be investigated within lowest time complexity. Let us assume that 'অধিকারী' (Odhikaari; owner) is our desired word. We can make-up the word by removing affix as per

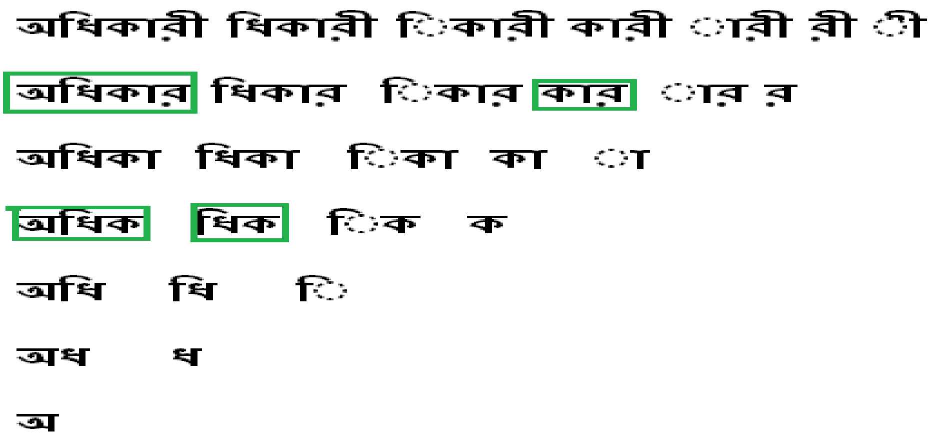

অধিকারী ধিকারী িকারী কারী ারী রী ী
অধিকার ধিকার িকার কার ার র
অধিকা ধিকা িকা কা া
অধিক ধিক িক ক
অধি ধি ি
অধ ধ
অ

Fig. 4: Arranging of a Word.

Here the 3 words such as ‘অধিকার (Odhikaar; rights)’, ‘অধিক (Odhik, more)’ and ‘কার (Kaar; whose)’ found in our root words corpus. But the lengthiest word is ‘অধিকার (Odhikaar; rights)’. So the desired root word of ‘অধিকারী (Odhikaari, owner))’ is ‘অধিকার (Odhikaar, rights)'. For the largest number of corpus words, we mapped the whole corpus.

3.3 Topic Modeling

In machine learning and natural language processing, a topic modeling is a type of statistical model for discovering the abstract ‘topics’ that occurs in a collection of documents [36]. Topic modeling is a frequently used text-mining tool for the discovery of hidden semantic structures in a text body [37]. Intuitively given that a document is about a particular topic, i. e., one would expect particular words to appear in the document more or less frequently. The ‘topics’ produced by topic modeling techniques are clusters of similar words. A topic modeling captures this intuition in a mathematical framework, which allows examining a set of documents and discovering, based on the statistics of the words in each, what the topics might be and what each document's balance of topics is. Topic modelings are also referred to as probabilistic topic models, which refer to statistical algorithms for discovering the latent semantic structures of an extensive text body [38].

Topic modeling is used in various different sections. There are major two sections describes below

A. Text classification – Topic modeling can improve classification by grouping similar words together in topics rather than using each word as a feature [39].

B. Recommender Systems – Using a similarity measure, we can build recommender systems. If our system recommend articles for readers, it will recommend articles with a topic structure similar to the articles the user has already read [40].

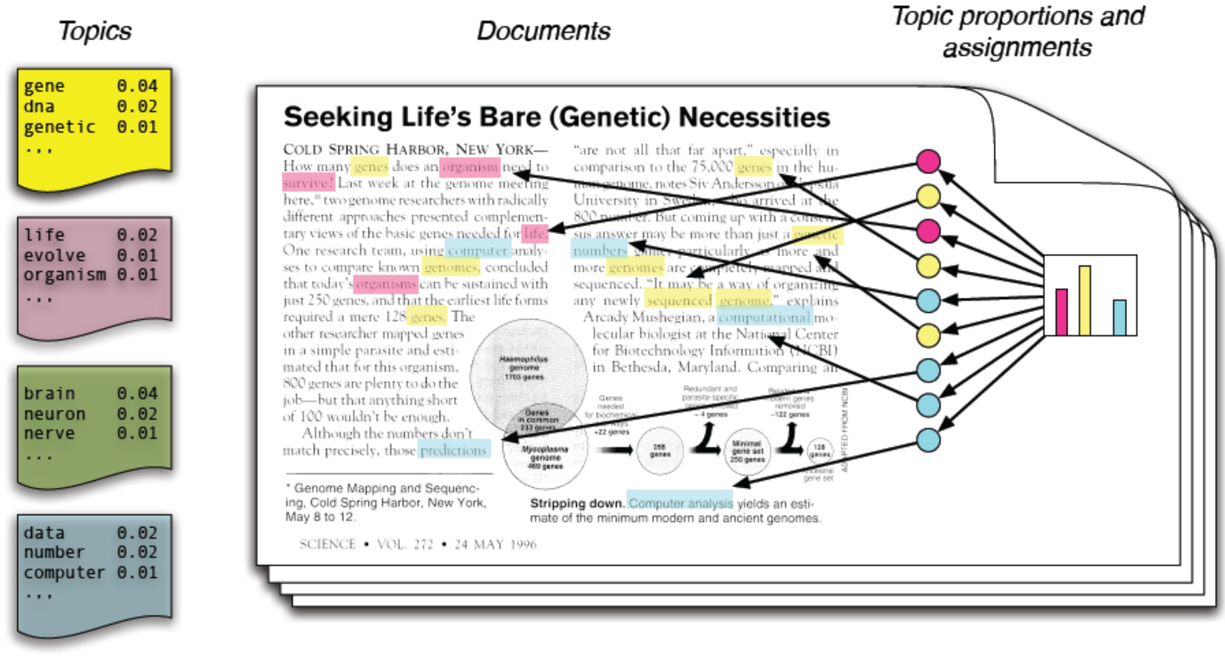


Fig.5: Topic Modeling

There are several algorithms for doing topic modeling. The most popular ones include

A. **LDA** – **Latent Dirichlet allocation** (**LDA**) is a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar. For example, if observations are words collected into documents, it posits that each document is a mixture of a small number of topics and that each word's presence is attributable to one of the document's topics. LDA is an example of a topic model.
B. **LSA or LSI** – Latent Semantic Analysis or Latent Semantic Indexing uses Singular Value Decomposition (SVD) on the Document-Term Matrix based on linear algebra [41].
C. **NMF** – NMF or Non-negative Matrix Factorization also known as non-negative matrix approximation is a group of algorithms in multivariate analysis and linear algebra where a matrix V is factorized into two matrices W and H with the property that all three matrices have no negative elements. This non-negativity makes the resulting matrices easier to inspect.

3.3.1 Non-negative Matrix Factorization

There are several ways in which the W and H may be found; Lee and Seung's multiplicative update rule has been a popular method due to simplicity of implementation. Let W and H non-negative. Then the update of the values in W and H by taking *n* as an index of the iteration is shown below.

$$H_{[i,j]}^{n+1} \leftarrow H_{[i,j]}^{n} \ \frac{((W^n)^T V)_{[i,j]}}{((W^n)^T) W^n H^n)_{[i,j]}}$$

and

$$W_{[i,j]}^{n+1} \leftarrow W_{[i,j]}^{n} \ \frac{(V(H^{n+1})^T)_{[i,j]}}{(W^n H^{n+1} (H^{n+1})^T)_{[i,j]}}$$

W and H are multiplicative factor identity matrix when V=WH.

NMF decomposes multivariate data by creating a user-defined number of features. Each feature is a linear combination of the original attribute set; the coefficients of these linear combinations are non-negative. NMF decomposes a data matrix V into the product of two lower rank matrices W and H so that V is approximately equal to W and times H. NMF uses an iterative procedure to modify the initial Values of W and H so that the product approaches V. The procedure terminates when the approximation error converges or the specified number of iterations is reached.

Let A=$\begin{bmatrix} 1 & 2 & 3 & 5 \\ 2 & 4 & 6 & 12 \\ 3 & 6 & 7 & 13 \end{bmatrix}$ is a matrix and also let 1st column indicates A1 and 3rd column indicates A3, similarly A2, A4 indicates 2nd and 4th column respectively.

Then by linear combination,

A1=1*A1+0*A3 A3=0*A1+1*A3

A2=2*A1+0*A3 A4=2*A1+1*A3

Then X=$\begin{bmatrix} 1 & 3 \\ 2 & 8 \\ 3 & 7 \end{bmatrix}$ and Y=$\begin{bmatrix} 1 & 2 & 0 & 2 \\ 0 & 0 & 1 & 1 \end{bmatrix}$, where X are many number of columns and Y are many number of rows.

X*Y=$\begin{bmatrix} 1 & 2 & 3 & 5 \\ 2 & 4 & 6 & 12 \\ 3 & 6 & 7 & 13 \end{bmatrix}$=A

3.4 Dimension Reduction

In machine learning classification problems, there are often too many factors on the basis of which the final classification is done. These factors are basically variables called features. The higher the number of features, the harder it gets to visualize the training set and then work on it. Sometimes, most of these features are correlated, and hence redundant. This is where dimensionality reduction algorithms come into play. Dimensionality reduction [45] is the process of reducing the number of random variables under consideration, by obtaining a set of principal variables. It can be divided into feature selection and feature extraction.

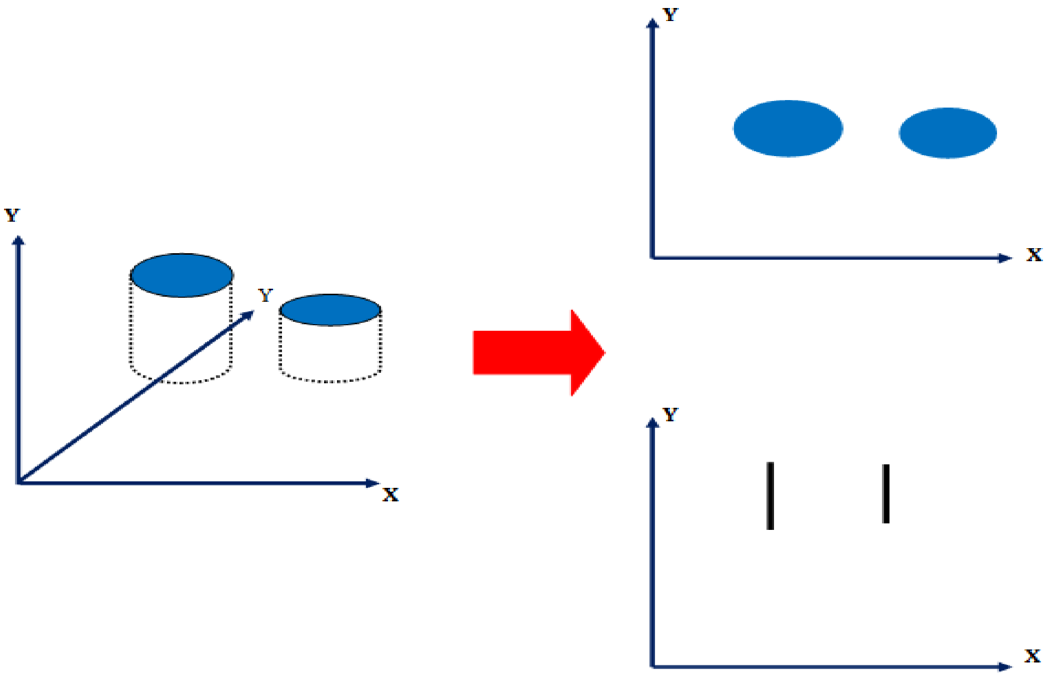


Fig.6: Converted 3-dimenstion to one dimension.

There are various algorithms Name for Dimensionality Reduction:

A. Genetic Algorithms (GA):

Genetic algorithms (GA)[46] are a broad class of algorithms that can be adapted to different purposes. They are *search algorithms* that are inspired by evolutionary biology and natural selection, combining mutation and cross-over to efficiently traverse large solution spaces.

B. Feature Extraction(FE):

Feature extraction [47] is for creating a new, smaller set of features that stills captures most of the useful information. Again, feature selection keeps a subset of the original features while feature extraction creates new ones.

C. Principal Component Analysis (PCA):

Principal component analysis (PCA) [48] is an unsupervised algorithm that creates linear combinations of the original features. The new features are orthogonal, which means that they are uncorrelated. Furthermore, they are ranked in order of their "explained variance."

D. Singular Value Decomposition

In linear algebra, the singular value decomposition *(*SVD*)* is a factorization of a real or complex matrix. It is the generalization of the Eigen decomposition [49] of a positive semi definite normal matrix (for example, a symmetric matrix with positive eigenvalues) to any matrix via an extension of the polar decomposition. It has many useful applications in signal processing and statistics.

3.4.1 Singular Value Decomposition

SVD is not totally algebraic as it is based on solving the minimization problem via Lagrange multipliers.

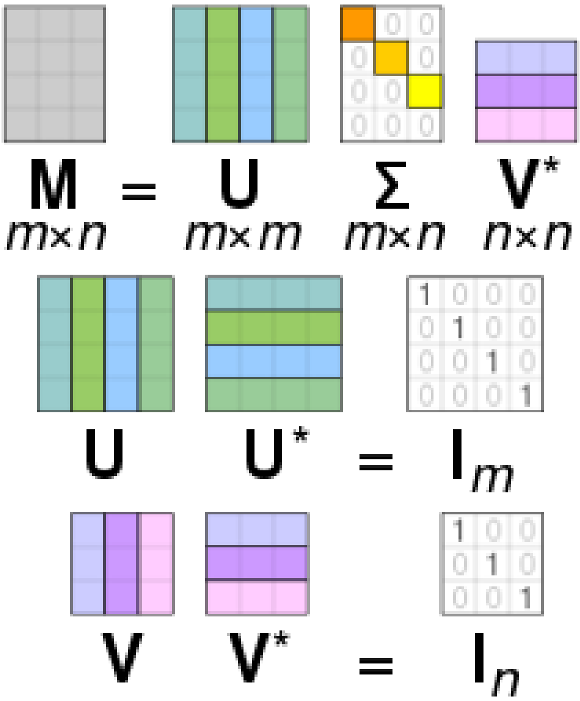


Fig.7: Dimension Reduction of SVD.

Formally the singular value decomposition of an m×n real or complex matrix M is a factorization of the form U∑V*, where U is an m×m real or complex unitary matrix, ∑ is an m×n rectangular diagonal matrix with non-negative real numbers on the diagonal, and V is an n×n real or complex unitary matrix. The diagonal entries ϭi of ∑ are known as the singular values of M. the columns of U and the columns of V are called the left-singular vectors and right-singular vectors of M respectively.

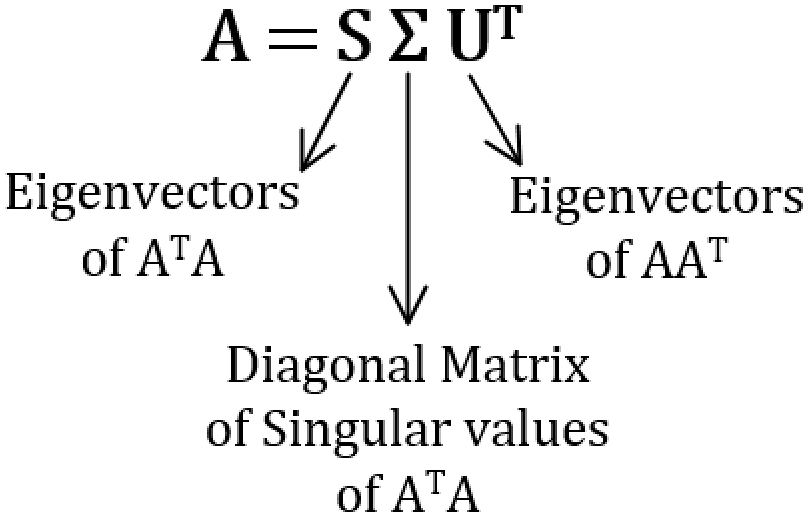


Fig.8: SVD Formula

More columns normally means more time required to build models and some data. If some columns have no predictive value, this means wasted time, or worsen, those columns contribute noise to the model and reduce model quality or predictive accuracy.

Dimensionality reduction can be achieved by simply dropping columns, for example, those that may show up as collinear with others or identified as not being particularly predictive of the target as determined by an attribute importance ranking technique. But it can also be achieved by deriving new columns based on linear combinations of the original columns. In both cases, the resulting transformed data set can be provided to machine learning algorithms to yield faster model build times, faster scoring times, and more accurate models. While SVD can be used for dimensionality reduction, it is often used tin digital signal processing for noise reduction, image compression, and other areas.

Consider C is a 2*2 matrix, the SVD of the matrix C is given below

$$C = \begin{bmatrix} 5 & 5 \\ -1 & 7 \end{bmatrix}$$

We know, $C = U\Sigma V^T$ (1)

Here, (1) $C^T C = U\Sigma^T \Sigma V^T$ (2)

(2) $CV = U\Sigma$ (3)

Now,

$$C^T C = \begin{pmatrix} 5 & -1 \\ 5 & 7 \end{pmatrix} \begin{pmatrix} 5 & 5 \\ -1 & 7 \end{pmatrix} = \begin{pmatrix} 26 & 18 \\ 18 & 74 \end{pmatrix}$$

This is a diagonalization of $C^T C$.

Now we need to find the eigenvalues which will be the entries of $\Sigma^T\Sigma$ and the eigenvectors which will be the columns of V.

For eigenvalues:

$$Det\ (\lambda I\ -\ C^T C) = \begin{vmatrix} \lambda - 26 & 18 \\ 18 & \lambda - 74 \end{vmatrix}$$

$$= (\lambda^2\ -\ 100\lambda\ +\ 1924) - 324$$

$$= (\lambda^2 - 100\lambda + 1600)$$

$$=\ (\lambda - 20)(\lambda - 80)$$

Therefore, the eigenvalues are 20 and 80.

For Eigenvectors:

When $\lambda\ =\ 20,$

$$C^T C - 20I = \begin{pmatrix} 6 & 18 \\ 18 & 54 \end{pmatrix}$$

$$V_1 = \begin{pmatrix} -\frac{3}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} \end{pmatrix}$$

When $\lambda\ =\ 80,$

$$C^T C - 20I = \begin{pmatrix} -54 & 18 \\ 18 & -6 \end{pmatrix}$$

$$V_2 = \begin{pmatrix} \frac{1}{\sqrt{10}} \\ \frac{3}{\sqrt{10}} \end{pmatrix}$$

Therefore, $V\ = (V_1 \quad V_2)$

$$V\ = \begin{pmatrix} -\frac{3}{\sqrt{10}} & \frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{3}{\sqrt{10}} \end{pmatrix}$$

Again $\sum\ =\ \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{pmatrix}$

Since $\sigma = \sqrt{\lambda}$, then $\sum = \begin{pmatrix} 2\sqrt{5} & 0 \\ 0 & 4\sqrt{5} \end{pmatrix}$

Now,

$$CV = \begin{pmatrix} 5 & 5 \\ -1 & 7 \end{pmatrix} \begin{pmatrix} \frac{-3}{\sqrt{10}} & \frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{3}{\sqrt{10}} \end{pmatrix}$$

$$= \begin{pmatrix} -\sqrt{10} & 2\sqrt{10} \\ \sqrt{10} & 2\sqrt{10} \end{pmatrix}$$

But from equation (3), we have

$$CV = U\Sigma$$

$$\Rightarrow U = CV\Sigma^{-1} \tag{4}$$

We know, $\Sigma^{-1} = \frac{1}{Det(\Sigma)} Adj(\Sigma)$

$$= \frac{1}{40} \begin{bmatrix} 4\sqrt{5} & 0 \\ 0 & 2\sqrt{5} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{2\sqrt{5}} & 0 \\ 0 & \frac{1}{4\sqrt{5}} \end{bmatrix}$$

Putting the value of Σ^{-1} in equation (4), we get

$$U = CV\Sigma^{-1}$$

$$= \begin{pmatrix} -\sqrt{10} & 2\sqrt{10} \\ \sqrt{10} & 2\sqrt{10} \end{pmatrix} \begin{pmatrix} \frac{1}{2\sqrt{5}} & 0 \\ 0 & \frac{1}{4\sqrt{5}} \end{pmatrix}$$

$$= \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

So the SVD of the given matrix can be shown as follows:

$$C = U\Sigma \mathrm{V}^T$$

$$\Rightarrow \begin{bmatrix} 5 & 5 \\ -1 & 7 \end{bmatrix} = \begin{pmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 2\sqrt{5} & 0 \\ 0 & 4\sqrt{5} \end{pmatrix} \begin{pmatrix} \frac{-3}{\sqrt{10}} & \frac{1}{\sqrt{10}} \\ \frac{1}{\sqrt{10}} & \frac{3}{\sqrt{10}} \end{pmatrix}$$

3.5 TF-IDF

TF-IDF is the abbreviation of the Term Frequency-Inverse Document Frequency. It is a numerical technique to find out the importance of a word to a sentence and is mathematical-statistically significant [50]. In this recipe of the TF-IDF model, there are few steps to find out the TF-IDF of an inserted sentence. Basically, TF determines the frequency of a term in a sentence. At first, to measure the value of TF of every pre-processed sentence, the term rule has been ensured.

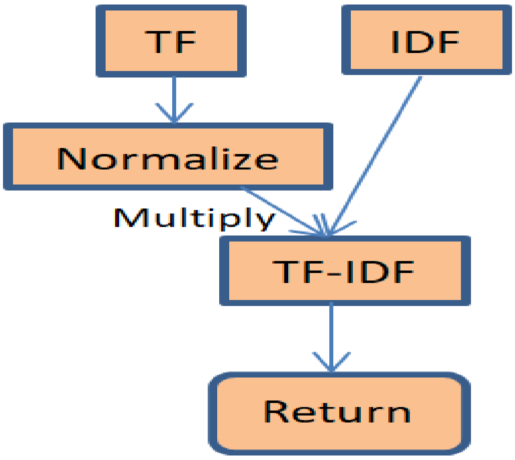


Fig.9: TF-IDF Processing.

For the standardization data, we normalize the whole term data on the same scale since small sentence’s frequency is neglected by a large one. We have followed the normalization technique as a term (word÷length) of the sentence.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}}$$

Here, $n_{i,j}$=i frequency in documents j

$\sum_k n_j$=Total words in documents j

Secondly, to discover a relevant sentence by searching questions, IDF keeps its effect hardly. In TF all the words are considered as equal importance. But IDF aids to determine the real importance of a word. To count the IDF we have followed this rule.

$$idf(w) = 1 + \log(\frac{N}{df_t})$$

Here, N=Total documents

df_t=Documents with term i

Finally, we count the desired TF-IDF by multiplication with the term TF and IDF from the inserted questions and sentences. In order to reduce the time and space complexity, we have calculated the TF-IDF only of those words which are related to inputted questions.

$$tf - idf = tf * idf$$

$$= \frac{n_{i,j}}{\sum_k n_{i,j}} * [1 + \log(\frac{N}{df_t})]$$

3.6 Cosine similarity

Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them [51].

The cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0,1] for any angle in the interval (0,π] radians. Such as

Cosine(degree)	0	30	45	60	90	120	180
Results	1	0.866	0.707	0.5	0	-0.5	-1

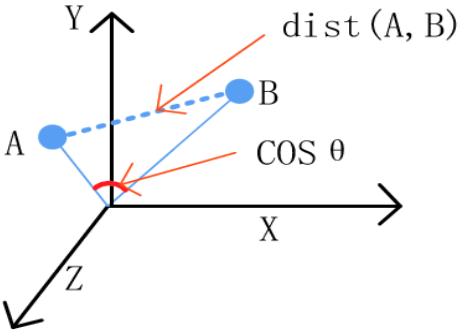


Fig. 10: Cosine Similarity.

In this project, cosine similarity is applied to ordain the relationship between questions. The cosine similarity is a measurement between two vectors that counts the cosine angle between them. The metric is a judgment of orientation but not magnitude that can be identified as a compare between vectors on a normalized space.

The cosine of two non-zero vectors can be derived as:

$$\vec{A}.\vec{B} = \left||A|\right|\left||B|\right| \cos\theta$$

$$or, \cos\theta = \frac{\vec{A}.\vec{B}}{\left||A|\right|\left||B|\right|}$$

$$= \frac{\sum_{i=1}^{n}(A_i * B_i)}{\sqrt{\sum_{i=1}^{n} {A_i}^2} * \sqrt{\sum_{i=1}^{n} {B_i}^2}}$$

$$so, similarity \ = Cosine(question, document)$$

$$= \frac{\sum_{i=1}^{n}(A_i * B_i)}{\sqrt{\sum_{i=1}^{n} {A_i}^2} * \sqrt{\sum_{i=1}^{n} {B_i}^2}}$$

3.7 Jaccard Similarity

The Jaccard index also called the Jaccard similarity coefficient is a statistic for comparing the similarity and diversity of sample sets which are distant. The Jaccard similarity measured the similarity between finite sample sets and is defined as the ratio between the size of the intersection and union of sample sets [52]. If P and Q be two sets the Jaccard index formula is given

$$J_{index}(P,Q) = \frac{|P \cap Q|}{|P \cup Q|} * 100 = \frac{|P \cap Q|}{|P|+|Q|-|P \cap Q|} * 100 \qquad \text{here, } 0 \leq J_{index}(P,Q) \leq 1$$

The Jaccard distance measures dissimilarity between sample sets that are complementary to the Jaccard coefficient.

$$J_{distance}(P,Q) = 1 - J_{index}(P,Q) * 100$$

Now, let an example using set notation and Venn-diagram

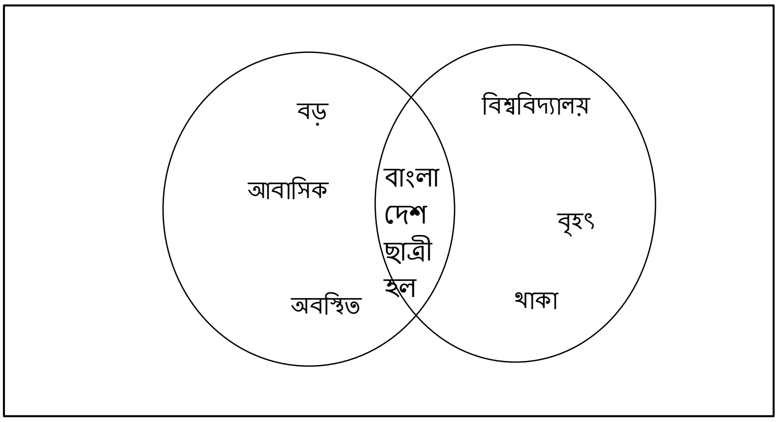


Fig.11: Jaccard Similarity.

P= {বাংলাদেশ, বড়, আবাসিক, ছাত্রী, হল, অবস্থিত}
Q= {বাংলাদেশ, বিশ্ববিদ্যালয়, বৃহৎ, ছাত্রী, হল, থাকা}

$$J_{index}(P,Q) = \frac{|P \cap Q|}{|P \cup Q|} * 100$$

$$= \frac{|\{\text{বাংলাদেশ, ছাত্রী, হল, }\}|}{|\{\text{ বাংলাদেশ, বড়, আবাসিক, ছাত্রী, হল, অবস্থিত, বিশ্ববিদ্যালয়, বৃহৎ, থাকা}\}|} * 100$$

$$= \frac{3}{9} * 100 = 33\%$$

And $J_{distance} = 1 - J_{index}\% = 1 - 33\% = 67\%$

So, we conclude four postulates

1. If $J_{index} \gg 50$ similar which means the sets share all members
2. If $J_{index} \ll 50$ similar which means the sets share few members
3. If $J_{index} \cong 50$ similar which means the sets share half of the members
4. If $J_{index} = 0$ similar which means the sets share no members.

3.8 Gradient Descent

Gradient Descent is an optimization algorithm used for minimizing the cost function in various machine learning algorithms. It is basically used for updating the parameters of the learning model.

Let $h_\theta(x)$ be the hypothesis for linear regression. Then, the cost function is given by: Let Σ represents the sum of all training examples from i=1 to m.

$J_{train}(\theta) = (1/2m)\, \Sigma(h_\theta(x^{(i)}) - y^{(i)})^2$

```
Repeat {
 $\theta j = \theta j$ – (learning rate/m) * $\Sigma( h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$
   For every j =0 …n
}
```

Where $x_j^{(i)}$ Represents the j^th^ feature of the i^th^ training example. So if *m* is very large, then the derivative term fails to converge at the global minimum.

3.9 Least square method

The method of least squares is a standard approach in regression analysis to approximate the solution of overdetermined systems, that is, sets of equations in which there are more equations than unknowns. Least squares mean that the overall solution minimizes the sum of the squares of the residuals made in the results of every single equation. Least square problems fall into two categories: Linear or ordinary least squares and nonlinear least squares, depending on whether or not the residuals are linear in all unknowns.

A more accurate way to find the equation of best fit for a set of ordered pairs $(x_1, y_1), (x_2, y_2), \dots \dots .(x_n, y_n)$ is

Step 1: Calculate the mean of the x-values and the mean of the y- values.

$$\bar{X} = \frac{\sum_{i=1}^{n} x_i}{n}$$

and

$$\bar{Y} = \frac{\sum_{i=1}^{n} y_i}{n}$$

Step 2: The following formula gives the slope of the line of best fit:

$$m = \frac{\sum_{i=1}^{n}(x_i - \bar{X})\,(y_i - \bar{Y})}{\sum_{i=1}^{n}(x_i - \bar{X})^2}$$

Step 3: Compute the y-intercept of the line by using the formula:

$$b = \bar{Y} - m\bar{X}$$

Step 4: Use the slope m and the y-intercept b to from the equation of the line.

Chapter Four

Proposed Work

Objective

- To illustrate proposal of the project
- To visualize graphical representation of proposed work

PROPOSED WORK

In this project paper, we present a Bengali Informative Intelligence Bot (BIIB) based on mathematics and statistics using Bengali Natural Language Processing (BNLP). The procedure is isolated in several parts that are: informative documents collection, pre-processing data, time and space complexity reduction , relationships between information & questions via the boost of TF-IDF model, Cosine Similarity and Jaccard Similarity. Corpus have been attached for the pre-processing inserted data. The action of Cosine Similarity is urged to obtain the relationship between the questions and answers. But Cosine Similarity deals with vectors. In this case, the documents and questions transmit to vectors using the TF-IDF model. In order to minimize the time and space complexity, we have urged the NMF & SVD techniques. Specially the Machine Learning procedure NMF deals with our methodology to reduce the time complexity and the matrix decomposition whereas SVD deals to reduce dimensions.

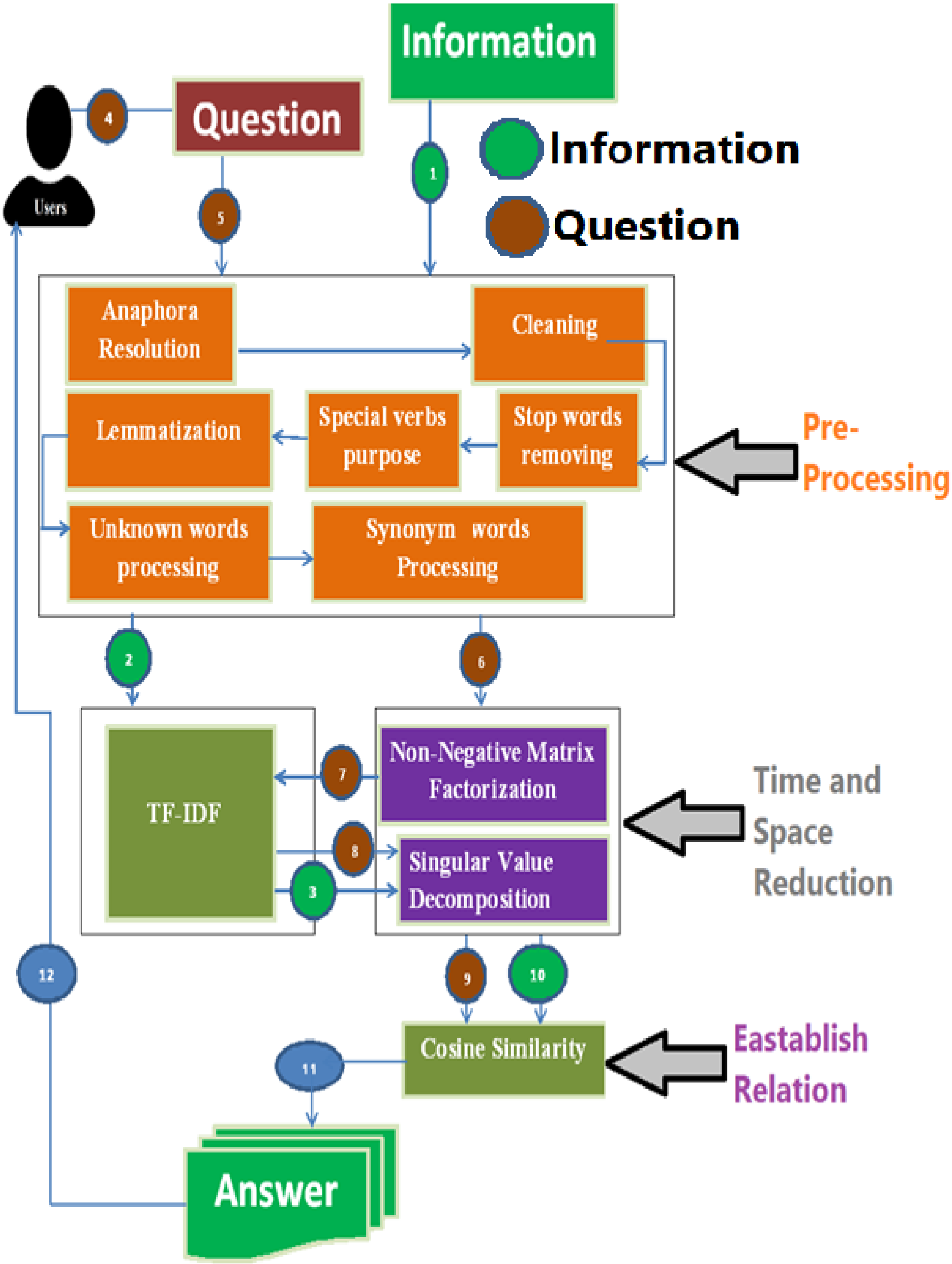


Fig.12: Proposed work-1

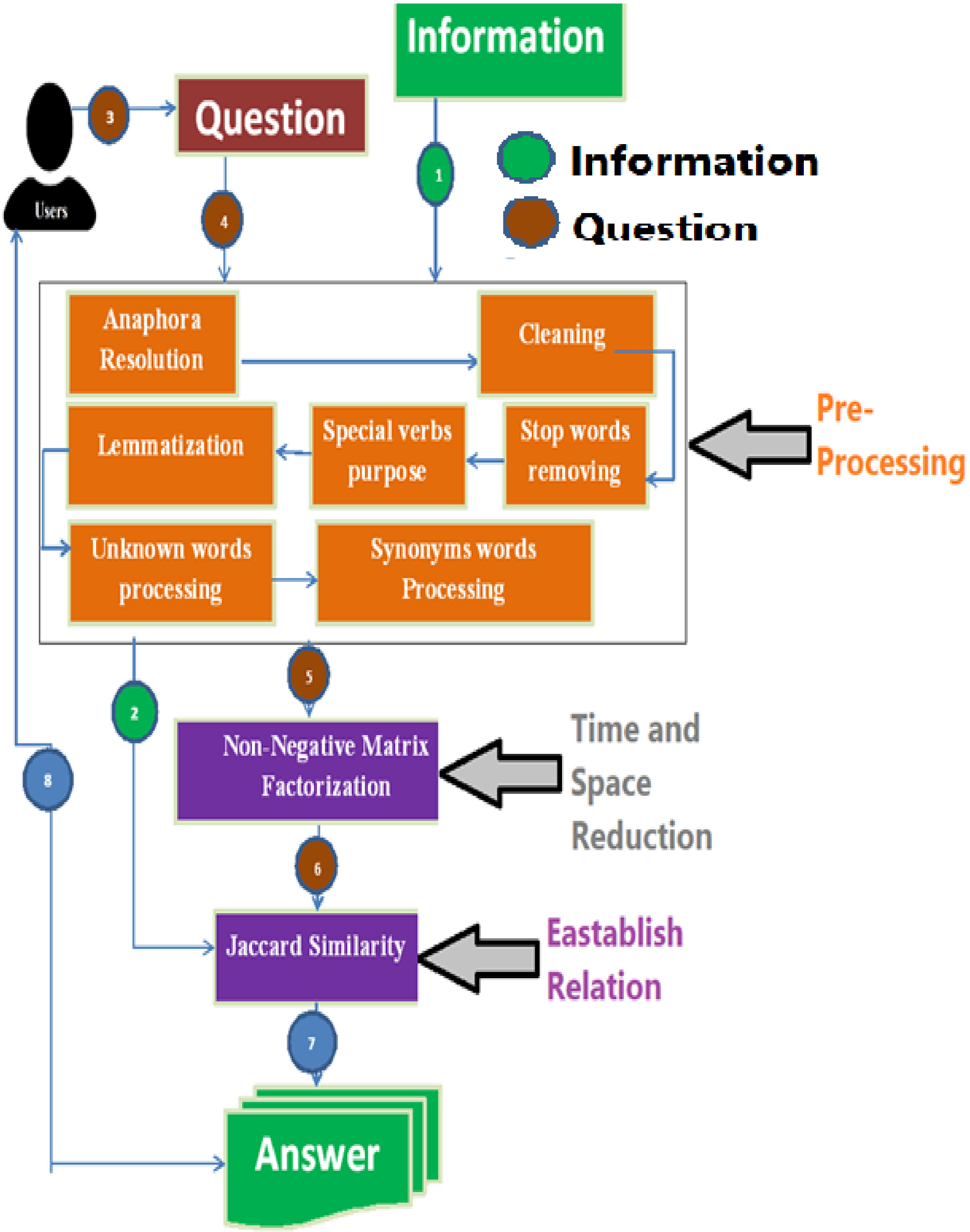
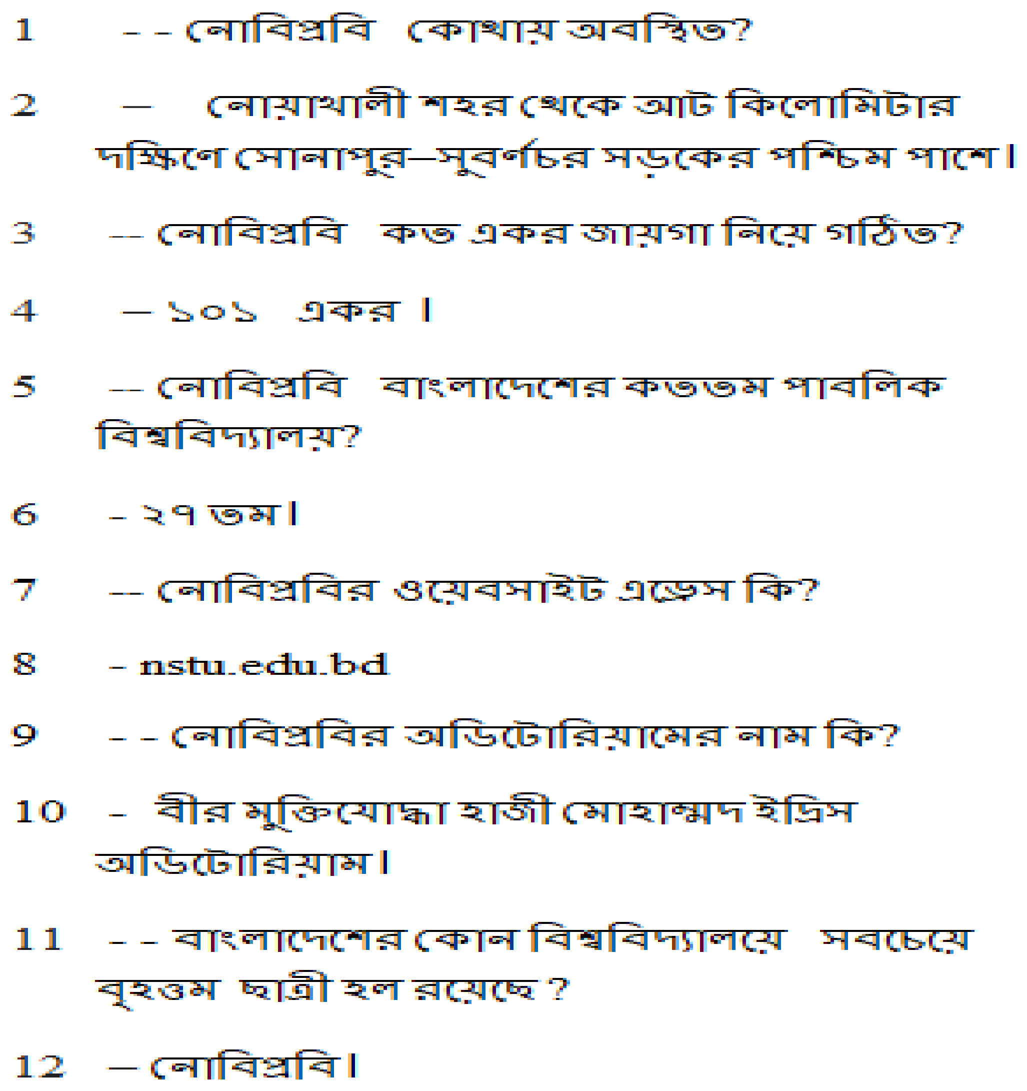


Fig.13: Proposed work-2.

For the visualization of our task, we introduce a simple informative corpus of NSTU and two relevant questions to find out the reply from this corpus.



1 - - নোবিপ্রবি কোথায় অবস্থিত?

2 — নোয়াখালী শহর থেকে আট কিলোমিটার দক্ষিণে সোনাপুর—সুবর্ণচর সড়কের পশ্চিম পাশে।

3 — নোবিপ্রবি কত একর জায়গা নিয়ে গঠিত?

4 — ১০১ একর ।

5 — নোবিপ্রবি বাংলাদেশের কততম পাবলিক বিশ্ববিদ্যালয়?

6 - ২৭ তম।

7 — নোবিপ্রবির ওয়েবসাইট এড্রেস কি?

8 - nstu.edu.bd

9 - - নোবিপ্রবির অডিটোরিয়ামের নাম কি?

10 - বীর মুক্তিযোদ্ধা হাজী মোহাম্মদ ইদ্রিস অডিটোরিয়াম।

11 - - বাংলাদেশের কোন বিশ্ববিদ্যালয়ে সবচেয়ে বৃহত্তম ছাত্রী হল রয়েছে ?

12 — নোবিপ্রবি।

Fig.14: Informative Question Dataset.

User Question-1**:** বাংলাদেশের সবচেয়ে বড় আবাসিক ছাত্রী হল কোথায় অবস্থিত?

User Question-2: এটির অডিটোরিয়ামের নাম কি?

Chapter Five

Pre-Processing

Objective

- To discuss relevant processes with examples
- To make these processes eligible to run on computer

PRE-PROCESSING

5.1 Anaphora

‘Anaphora’ refers to a word that is used earlier in a sentence to avoid repetition, e. g., the pronouns. It requires a successful identification and resolution of NLP. In the proposed BIIB, we describe a review of work done in the field of anaphora resolution which has an influence on pronouns, mainly personal pronoun. The Hobbs’ algorithm of Anaphora Resolution is used in our proposed BIIB. The algorithm has been adapted for Bengali language taking into account the roles of subject, object, and its impact on Anaphora Resolution for reflexive and possessive pronouns.

বাংলাদেশের	সবচেয়ে	বড়	আবাসিক	ছাত্রী	হল	কোথায়	অবস্থিত?
নোবিপ্রবিতে	অবস্থিত।						
এটির	অডিটোরিয়ামের	নাম	কি?				
বীর	মুক্তিযোদ্ধা	হাজী	মোহাম্মদ	ইদ্রিস	অডিটোরিয়াম।		

Table 1: Workflow of Anaphora Resolution.

Here, ‘এটি’ (It) is the pronoun of the name of ‘নোবিপ্রবি’(NSTU). So we used ‘এটি’ instead of ‘নোবিপ্রবি’ from the previous example as Anaphora Resolution.

User Question-1: বাংলাদেশের সবচেয়ে বড় আবাসিক ছাত্রী হল কোথায় অবস্থিত?

BIIB Answer-1**:** নোবিপ্রবিতে অবস্থিত।

User Question-2: নোবিপ্রবির অডিটোরিয়ামের নাম কি?

BIIB Answer-2: বীর মুক্তিযোদ্ধা হাজী মোহাম্মদ ইদ্রিস অডিটোরিয়াম।

5.2 Cleaning Punctuations

Cleaning word refers to remove an unwanted character which does not have any sentiment on informative data; for example: colon, semicolon, comma, question mark, exclamation point, and other punctuations. We urge the BRE tool to shift the unwanted characters from the information and questions. The Bengali punctuation corpus has been edified to deal with BRE and to assist to carry away punctuation as unwanted data.

User Questions-1: বাংলাদেশের সবচেয়ে বড় আবাসিক ছাত্রী হল কোথায় অবস্থিত

User Questions-2: নোবিপ্রবির অডিটোরিয়ামের নাম কি

5.3 Stop Words Removing

Stop words refer the words that does not have any influence on documents or sentences. Instances of Bengali stop words are এবং (and), কোথায় (where), অথবা (or), তে (to), সাথে (with) etc. Since our BIIB is an algorithm based data, the stop words need to be dismissed. In our inserted data, every word is checked, that is, words contained in Bengali stop words corpus or not are deleted. For the simplification of this action, we have used BLTK tools that we made.

User Questions-1: বাংলাদেশের বড় আবাসিক ছাত্রী হল অবস্থিত

User Questions-2: নোবিপ্রবির অডিটোরিয়ামের নাম

Here we have removed the Bengali stop words ‘সবচেয়ে (most)’, ‘কোথায় (where)’ and ‘কি (what)’ from the previous questions.

5.4 Verb Processing

In BNLP, there are few verbs that cannot be lemmatized by any system because of ignoring all kinds of lemmatization algorithms. For example, (গেলে, went) and (গিয়ে, going) generate from the root word (যাওয়া, go). There are no relations of character between (গেলে, went) and (যাওয়া, go). So processing with algorithms to these words is not good choice. That is why these types of verbs are converted into their root verbs for easily accessing as lemmatization.

5.5 Lemmatization

In our project paper, we used different kinds of lemmatization techniques like DBSRA and Trie. Sometimes there are few words in Bengali language which do not work in Trie but work in DBSRA or work in Trie but not in DBSRA. So we have used Levenshtein distance to find out the best lemma word between DBSRA and trie.

Sometimes the lemmatization algorithms are not a good choice for the unknown words. Here unknown words refer to the name of a place, person or name of anything. Levenshtein distance assists to determine which word is known or unknown. We count the probability of edit between lemma and word (before lemmatization). If the probability P (lemma |word) is greater than 50% [P (lemma|word) > 50%], then it is counted as unknown words.

In order to process the unknown words, we have established a corpus of the suffix of Bengali language; for instances: তে (te), ছে(che), য়ের(yer) etc. The longest common suffix has been removed from the last position of an unknown word. Thus we obtain the lemma or root of an unknown word.

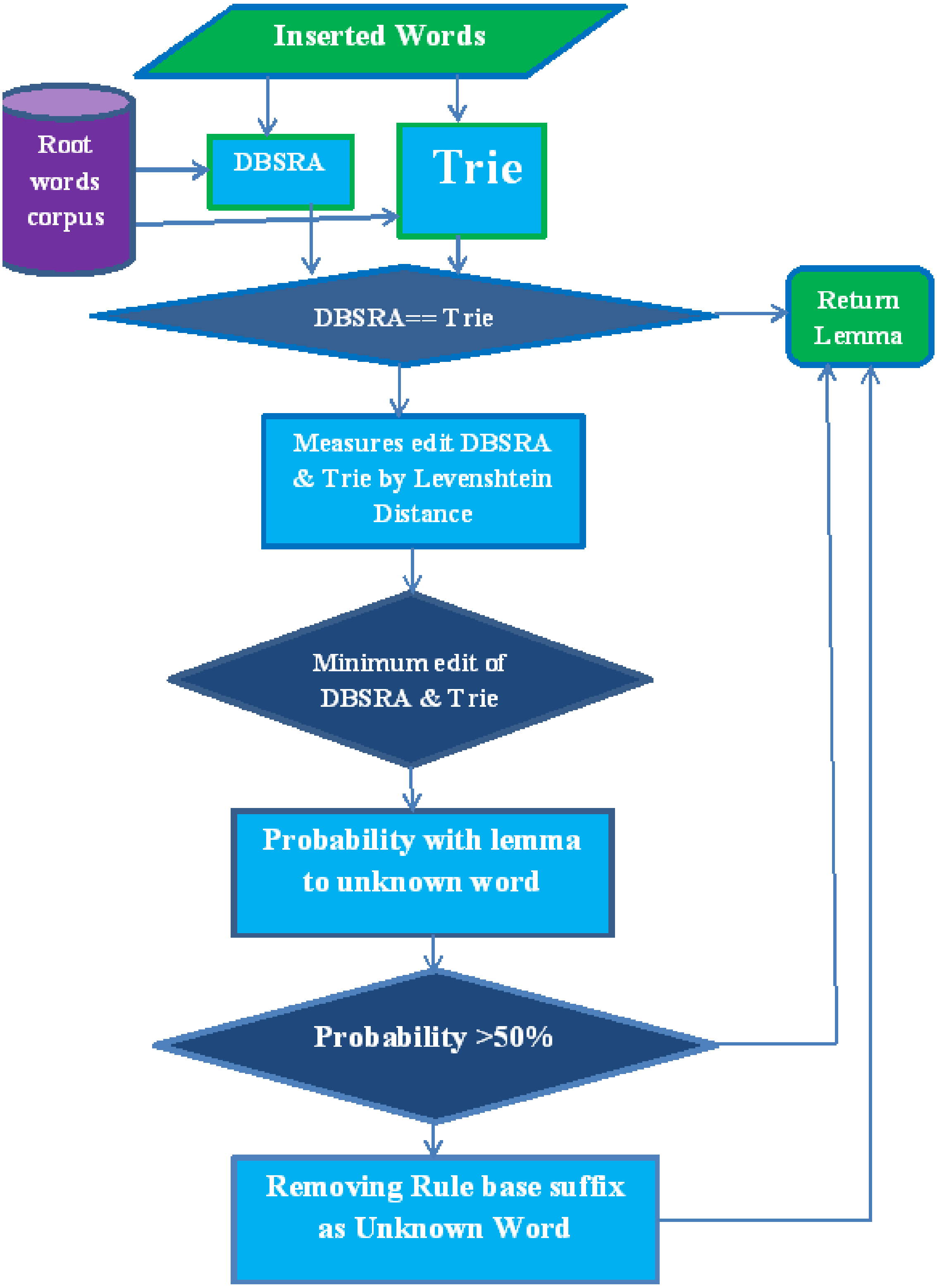

Inserted Words
Root words corpus
DBSRA
Trie
DBSRA== Trie
Return Lemma
Measures edit DBSRA & Trie by Levenshtein Distance
Minimum edit of DBSRA & Trie
Probability with lemma to unknown word
Probability >50%
Removing Rule base suffix as Unknown Word

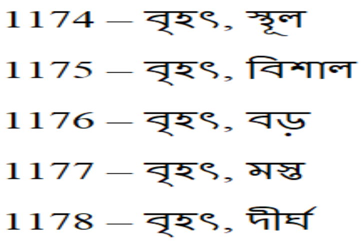
Fig.15: Lemmatization process.

User Questions-1: বাংলাদেশ বড় আবাসিক ছাত্রী হল অবস্থিত

User Questions-2: নোবিপ্রবি অডিটোরিয়াম নাম

5.6 Synonym Words Processing

Synonym words indicate the exact or nearly same meaning of different words. Users ask questions having words which are not available in information data but its meanings do. In this sense, BIIB may fail to answer correctly in many ways. So synonym words processing have a significant action in the BIIC as well as in Natural Language Understanding (NLU). To recover this bad situation, a synonym word corpus has been constructed containing total 13,189 words. Every word is mapped to a common word. In the BIIB processing, if a word is not in synonym words corpus, it does not need to take action as a similar word.



1174 — বৃহৎ, স্থূল
1175 — বৃহৎ, বিশাল
1176 — বৃহৎ, বড়
1177 — বৃহৎ, মস্ত
1178 — বৃহৎ, দীর্ঘ

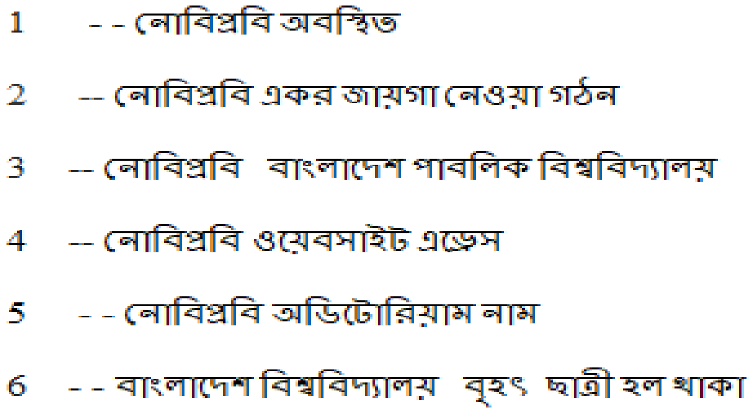
Fig.16: A part of synonym words Dataset.

User Question-1: বাংলাদেশ বৃহৎ আবাসিক ছাত্রী হল অবস্থিত

User Question-2: নোবিপ্রবি অডিটোরিয়াম নাম

Here ‘বৃহৎ (large)’ is the synonym of ‘বড় (big)’ and ‘বৃহৎ’(large) is considered as a common word.

Till now we have showed our project using only two users questions. Similarly we get from the considered informative corpus and users questions after pre-processing as follows:



1 - - নোবিপ্রবি অবস্থিত
2 -- নোবিপ্রবি একর জায়গা নেওয়া গঠন
3 -- নোবিপ্রবি বাংলাদেশ পাবলিক বিশ্ববিদ্যালয়
4 -- নোবিপ্রবি ওয়েবসাইট এড্রেস
5 - - নোবিপ্রবি অডিটোরিয়াম নাম
6 - - বাংলাদেশ বিশ্ববিদ্যালয় বৃহৎ ছাত্রী হল থাকা

Fig.17: Pre-processed Data.

Chapter Six

Time and Space Complexity Reduction

Objective

- To reduce time complexity with NMF
- To reduce space complexity with SVD

TIME AND SPACE COMPLEXITY REDUCTION

6.1 Topic Classification

If a user ask BIIB a question, then it has to reply after searching from whole topics. It takes a lot of time to find out the desired answer, and the time complexity increases. But time complexity can be minimized when BIIB can find out the most related topics easily. To get instant reply, we use topic classification.

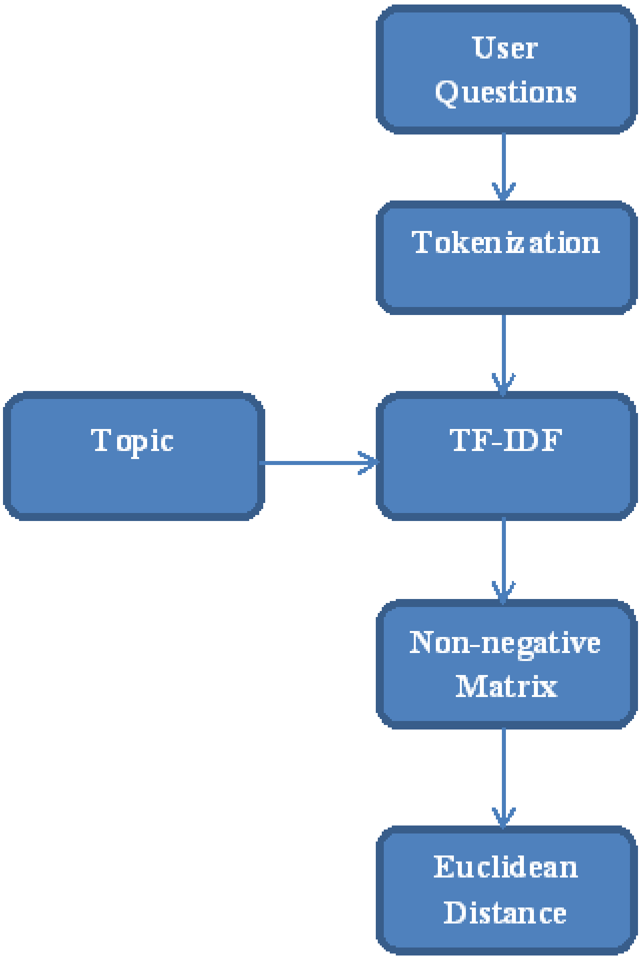


Fig. 18: Methodology of Topic Classification.

The methodology of topic classifications works as follows:

i) Every word of the user questions needs to be tokenized.

ii) TF-IDF converts the words into their numerical value according to their sentiment.

iii) NMF separates the numerical values into two non-negative matrices for easier calculation.

iv) Euclidean distance traces the distance between user question and topic, and finds the probability.

In our proposed work, we have collected 74 different topics related information of NSTU. For the simplicity of calculation, we now consider about 2 topics; Topic-1 relates information about NSTU and Topic-2 about departments. Both the considered topics contain their respective influential words.

Topic-1 (NSTU): নোবিপ্রবি, অডিটোরিয়াম, ওয়েবসাইট, বিশ্ববিদ্যালয়, হল

Topic-2 (Department): নোবিপ্রবি, ডিপার্টমেন্ট, ক্লাস, চেয়ারম্যান, টিচার

From the previous chapter, we get the pre-processed users

User Question-1: বাংলাদেশ বৃহৎ আবাসিক ছাত্রী হল অবস্থিত

User Question-2: নোবিপ্রবি অডিটোরিয়াম নাম

Terms	Topics		IDF	TF*IDF	
	T-1	T-2		T-1	T-2
নোবিপ্রবি	1/5	1/5	1+log(2/2)=1	0.2	0.2
অডিটোরিয়াম	1/5	0	1+log(2/1)=1.301	0.2602	0
ওয়েবসাইট	1/5	0	1+log(2/1)=1.301	0.2602	0
টিচার	0	1/5	1+log(2/1)=1.301	0	0.2602
বিশ্ববিদ্যালয়	1/5	0	1+log(2/1)=1.301	0.2602	0
হল	1/5	0	1+log(2/1)=1.301	0.2602	0
ডিপার্টমেন্ট	0	1/5	1+log(2/1)=1.301	0	0.2602
চেয়ারম্যান	0	1/5	1+log(2/1)=1.301	0	0.2602
ক্লাস	0	1/5	1+log(2/1)=1.301	0	0.2602

Table 2: TF-IDF of topics for topic classification

Terms	User Questions		IDF	TF*IDF	
	Q-1	Q-2		Q-1	Q-2
নোবিপ্রবি	0	1/3	1+log(2/2)=1	0	0.333
অডিটোরিয়াম	0	1/3	1+log(2/1)=1.301	0	0.433
ওয়েবসাইট	0	0	1+log(2/1)=1.301	0	0
টিচার	0	0	1+log(2/1)=1.301	0	0
বিশ্ববিদ্যালয়	0	0	1+log(2/1)=1.301	0	0
হল	1/6	0	1+log(2/1)=1.301	0.217	0
ডিপার্টমেন্ট	0	0	1+log(2/1)=1.301	0	0
চেয়ারম্যান	0	0	1+log(2/1)=1.301	0	0
ক্লাস	0	0	1+log(2/1)=1.301	0	0

Table 3: TF-IDF of user questions for topic classification.

Considering the TF-IDF from Table- 2 as V (non-negative matrix), we can factorize it into two non-negative matrices W and H as follows:

$$V = \begin{bmatrix} 0.2 & 0.2 \\ 0.2602 & 0 \\ 0.2602 & 0 \\ 0 & 0.2602 \\ 0.2602 & 0 \\ 0.2602 & 0 \\ 0 & 0.2602 \\ 0 & 0.2602 \\ 0 & 0.2602 \end{bmatrix}$$

$$= \begin{bmatrix} 0.384055 & 0.279394 \\ 0.499655 & 0 \\ 0.499655 & 0 \\ 0 & 0.363513 \\ 0.499655 & 0 \\ 0.499655 & 0 \\ 0 & 0.363513 \\ 0 & 0.363513 \\ 0 & 0.363513 \end{bmatrix} \begin{bmatrix} 0.520759 & 0 \\ 0 & 0.715797 \end{bmatrix}$$

$$= WH$$

Where,

$$W = \begin{bmatrix} 0.34055 & 0.279394 \\ 0.499655 & 0 \\ 0.499655 & 0 \\ 0 & 0.363513 \\ 0.499655 & 0 \\ 0.499655 & 0 \\ 0 & 0.363513 \\ 0 & 0.363513 \\ 0 & 0.363513 \end{bmatrix}, \qquad H = \begin{bmatrix} 0.520759 & 0 \\ 0 & 0.715797 \end{bmatrix}$$

[There are many popular models to factorize non-negative matrix (NMF). Two of them are: Least square method and gradient descent method. We have used the gradient descent method to factorize V into W and H.]

Now we know that

$$q_1 = q_1^T W$$

$$\therefore q_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0.217 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.34055 & 0.279394 \\ 0.499655 & 0 \\ 0.499655 & 0 \\ 0 & 0.363513 \\ 0.499655 & 0 \\ 0.499655 & 0 \\ 0 & 0.363513 \\ 0 & 0.363513 \\ 0 & 0.363513 \end{bmatrix}$$

$$q_1 = \begin{bmatrix} 0.108425 & 0 \end{bmatrix}$$

Similarly, $q_2 = q_2^T W$

$$\therefore q_2 = \begin{bmatrix} 0.333 & 0.333 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0.34055 & 0.279394 \\ 0.499655 & 0 \\ 0.499655 & 0 \\ 0 & 0.363513 \\ 0.499655 & 0 \\ 0.499655 & 0 \\ 0 & 0.363513 \\ 0 & 0.363513 \\ 0 & 0.363513 \end{bmatrix}$$

$$= \begin{bmatrix} 0.279788 & 0.093038 \end{bmatrix}$$

Rows of H hold eigenvector values. These are the coordinates of individual document vectors.

Hence, $T_1 = (H_{11}, H_{12}) = (0.52079, 0)$

$T_2 = (H_{21}, H_{22}) = (0, 0.715797)$

Now we will find the Euclidean distances between user questions and topics.

$$dis(q_1, T_1) = \sqrt{(0.108425 - 0.520759)^2 + (0 - 0)^2}$$

$$= 0.412334$$

$$dis(q_1, T_2) = \sqrt{(0.108425 - 0)^2 + (0 - 0.715797)^2}$$

$$= 0.723962$$

Since, $dis(q_1, T_1) < dis(q_1, T_2)$, so user question-1 is related to topic-1.

Similarly,

$$dis(q_2, T_1) = \sqrt{(0.279788 - 0.520759)^2 + (0.093038 - 0)^2}$$

$$= 0.258308$$

$$dis(q_2, T_2) = \sqrt{(0.279788 - 0)^2 + (0.093038 - 0.715797)^2}$$

$$= 0.682723$$

Since, $dis(q_2, T_1) < dis(q_2, T_2)$, so user question-2 is related to topic-1.

6.2 Dimension Reduction

In order to represent the words of user questions or informative questions, we have used the vectorization method TF-IDF model. For the standardization data, we normalize the whole term data on the same scale since small sentence's frequency is neglected by a large one. We have followed the normalization technique as a term (word ÷ length) of the sentence.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}}$$

For the simplification for our task, we have taken two examples from the considered corpus. The TF-IDF value of the informative questions and user question have been shown in the table.

Terms	Word		IDF	TF*IDF	
	IQ-1	IQ-2		IQ-1	IQ-2
নোবিপ্রবি	1/3	0	1+log(2/1)=1.301	0.4337	0
অডিটোরিয়াম	1/3	0	1+log(2/1)=1.301	0.4337	0
নাম	1/3	0	1+log(2/1)=1.301	0.4337	0
বাংলাদেশ	0	1/6	1+log(2/1)=1.301	0	0.217
বিশ্ববিদ্যালয়	0	1/6	1+log(2/1)=1.301	0	0.217
বৃহৎ	0	1/6	1+log(2/1)=1.301	0	0.217
ছাত্রী	0	1/6	1+log(2/1)=1.301	0	0.217
হল	0	1/6	1+log(2/1)=1.301	0	0.217
থাকা		1/6	1+log(2/1)=1.301	0	0.217

Table 4: TF-IDF value for informative questions.

Terms	Word		IDF	TF*IDF	
	Q-1	Q-2		Q-1	Q-2
নোবিপ্রবি	0	1/3	1+log(2/1)=1.301	0	0.4337
অডিটোরিয়াম	0	1/3	1+log(2/1)=1.301	0	0.4337
নাম	0	1/3	1+log(2/1)=1.301	0	0.4337
বাংলাদেশ	1/6	0	1+log(2/1)=1.301	0.217	0
বিশ্ববিদ্যালয়	0	0	1+log(2/1)=1.301	0	0
বৃহৎ	0	0	1+log(2/1)=1.301	0	0
ছাত্রী	1/6	0	1+log(2/1)=1.301	0.217	0
হল	1/6	0	1+log(2/1)=1.301	0.217	0
থাকা	0	0	1+log(2/1)=1.301	0	0

Table 5: TF-IDF value for users questions.

There are few dimension reduction techniques in Machine Learning like as GA, FE, PCA, SVD etc. Among these techniques, SVD is used in our research project for its mathematical qualification.

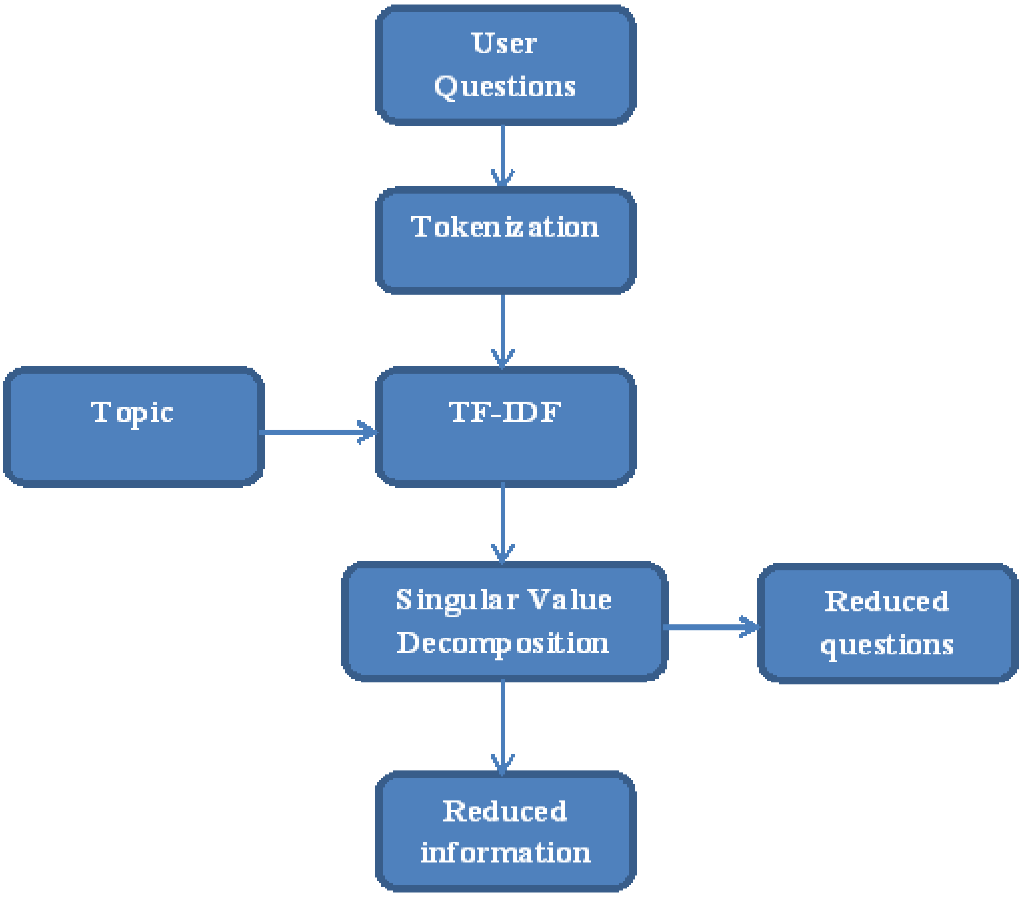


Fig.19: Methodology of Dimension Reduction.

Now, we will show the size or dimension reduction from the previous TF-IDF value of our considered problems.

Terms	Informative Questions (IQ)		Users Questions (q)	
	IQ_1	IQ_2	q_1	q_2
নোবিপ্রবি	0.4337	0	0	0.4337
অডিটোরিয়াম	0.4337	0	0	0.4337
নাম	0.4337	0	0	0.4337
বাংলাদেশ	0	0.217	0.217	0
বিশ্ববিদ্যালয়	0	0.217	0	0
বৃহৎ	0	0.217	0.217	0
ছাত্রী	0	0.217	0.217	0
হল	0	0.217	0.217	0
থাকা	0	0.217	0	0

Table 6: Matrix of Informative Questions and User Questions.

Let us set the term weights and construct the term-document matrix C and query matrix:

$$C = \begin{bmatrix} 0.4337 & 0 \\ 0.4337 & 0 \\ 0.4337 & 0 \\ 0 & 0.217 \\ 0 & 0.217 \\ 0 & 0.217 \\ 0 & 0.217 \\ 0 & 0.217 \\ 0 & 0.217 \end{bmatrix}, \qquad q_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.217 \\ 0 \\ 0.217 \\ 0.217 \\ 0.217 \\ 0 \end{bmatrix}, \qquad q_2 = \begin{bmatrix} 0.4337 \\ 0.4337 \\ 0.4337 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Now we will use SVD in matrix C and will find $U,\ \Sigma$ and V matrices, where

$$C = U\Sigma V^T$$

$$C = \begin{bmatrix} -0.57735 & 0 \\ -0.57735 & 0 \\ -0.57735 & 0 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \end{bmatrix} \begin{bmatrix} 0.75119 & 0 \\ 0 & 0.53153 \end{bmatrix} \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}^T$$

Here, the singular value $\sigma_1 = 0.75119$ are not too different from $\sigma_2 = 0.53153$. So we cannot avoid σ_2. If σ_1>>σ_2 , then we could avoid σ_2 and it would reduce the singular value matrix size into one dimension.

Hence the dimension of U is 2×9, so $k = 2$.

Now let us consider

$$U \approx U_k = \begin{bmatrix} -0.57735 & 0 \\ -0.57735 & 0 \\ -0.57735 & 0 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \end{bmatrix}, \qquad \Sigma \approx \Sigma_k = \begin{bmatrix} 0.75119 & 0 \\ 0 & 0.53153 \end{bmatrix}$$

$$V \approx V_k = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \qquad V^T \approx V_k^T = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$$

Rows of V hold the eigenvector values. These are the co-ordinates of individual document vectors. Hence

$IQ_1 = (-1{,}0)$ and $IQ_2 = (0,-1)$

To find the new query vector co-ordinates, we have

$$q = q^T U_k \Sigma_k^{-1}$$

Therefore,

$$q_1 = q_1^T U_k \Sigma_k^{-1}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0.217 & 0 & 0.217 & 0.217 & 0.217 & 0 \end{bmatrix} \begin{bmatrix} -0.57735 & 0 \\ -0.57735 & 0 \\ -0.57735 & 0 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \end{bmatrix} \begin{bmatrix} \frac{1}{0.75119} & 0 \\ 0 & \frac{1}{0.53153} \end{bmatrix}$$

$$q_1 = \begin{bmatrix} 0 & \frac{-11073727}{66442375} \end{bmatrix}$$

Similarly, $q_2 = q_2^T U_k \Sigma_k^{-1}$

$$=\begin{bmatrix}0.4337 & 0.4337 & 0.4337 & 0 & 0 & 0 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}-0.57735 & 0 \\ -0.57735 & 0 \\ -0.57735 & 0 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824 \\ 0 & -0.40824\end{bmatrix}\begin{bmatrix}\frac{1}{0.75119} & 0 \\ 0 & \frac{1}{0.53153}\end{bmatrix}$$

$$q_2 = \begin{bmatrix}\frac{-150238017}{150238000} & 0\end{bmatrix}$$

Chapter Seven

Establishment of Relationship

Objective

- To find the relation between user questions & answers with Cosine similarity and Jaccard similarity

ESTABLISHMENT OF RELATIONSHIP

7.1 Cosine Similarity

After reducing the dimension of the informative questions and user questions, we get from the last part of the previous chapter

$$IQ_1 = (-1{,}0) \text{ and } IQ_2 = (0, -1)$$

$$q_1 = \begin{bmatrix} 0 & \frac{-11073727}{66442375} \end{bmatrix}$$

$$q_2 = \begin{bmatrix} \frac{-150238017}{150238000} & 0 \end{bmatrix}$$

Now we will find the cosine similarities.

$$\cos \theta_1 = \mathrm{sim}(\mathrm{q}_1, \mathrm{IQ}_1) = \frac{\mathrm{q}_1 \cdot \mathrm{IQ}_1}{|\mathrm{q}_1|\,|\mathrm{IQ}_1|}$$

$$= \frac{(0)(-1)+(-\frac{11073727}{66442375})(0)}{\sqrt{(0)^2+(-\frac{11073727}{66442375})^2}\sqrt{(-1)^2+(0)^2}}$$

$$= 0$$

Similarly,

$$\cos \theta_2 = sim(q_1, IQ_2) = \frac{q_1 \cdot IQ_2}{|q_1|\,|IQ_2|}$$

$$= \frac{(0)(0)+(-\frac{11073727}{66442375})(-1)}{\sqrt{(0)^2+(-\frac{11073727}{66442375})^2}\sqrt{(0)^2+(-1)^2}}$$

$$= 1$$

We can see that $sim(q_1, IQ_2) > sim(q_1, IQ_1)$. So users question1 can be found in informative question2, i.e., IQ_2.

Again,

$$\cos\theta_3 = sim(q_2, IQ_1) = \frac{q_2 \cdot IQ_1}{|q_2|\,|IQ_1|}$$

$$= \frac{\left(-\frac{150238017}{150238000}\right)(-1)+(0)(0)}{\sqrt{(-\frac{150238017}{150238000})^2+(0)^2}\sqrt{(-1)^2+(0)^2}}$$

$$= 1$$

And,

$$\cos\theta_4 = sim(q_2, IQ_2) = \frac{q_2 \cdot IQ_2}{|q_2|\,|IQ_2|}$$

$$= \frac{\left(-\frac{150238017}{150238000}\right)(0)+(0)(-1)}{\sqrt{(-\frac{150238017}{150238000})^2+(0)^2}\sqrt{(0)^2+(-1)^2}}$$

$$= 0$$

Since, $sim(q_2, IQ_1) > sim(q_2, IQ_2)$, so users question-2 can be found in informative question1, i.e., IQ_1.

7.2 Jaccard Similarity

From the Chapter 5, we get

q_1 = বাংলাদেশ বড় আবাসিক ছাত্রী হল অবস্থিত

q_2 = নোবিপ্রবি অডিটোরিয়াম নাম

IQ_1 = নোবিপ্রবি অডিটোরিয়াম নাম

IQ_2 = বাংলাদেশ বিশ্ববিদ্যালয় বৃহৎ ছাত্রী হল থাকা

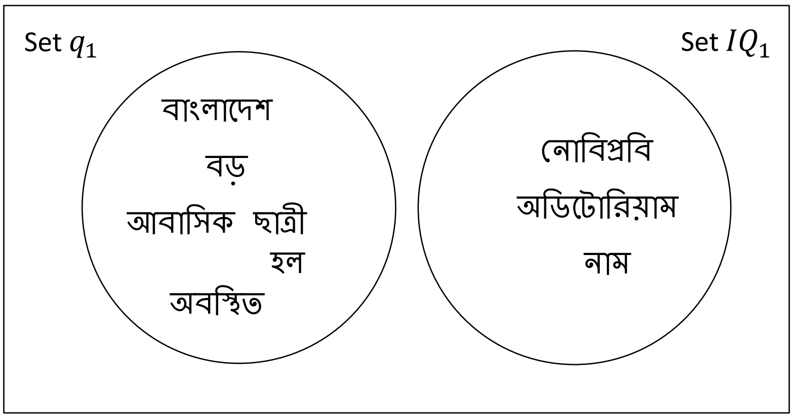
$\therefore$ q_1 has 6 elements

q_2 has 3 elements

IQ_1 has 3 elements

IQ_2 has 6 elements

(i) Now we will find the Jaccard similarity between set q_1 and set IQ_1.



From the Venn diagram, we see that

$q_1 \cap IQ_1$ has 0 elements

$q_2 \cup IQ_1$ has 0 elements

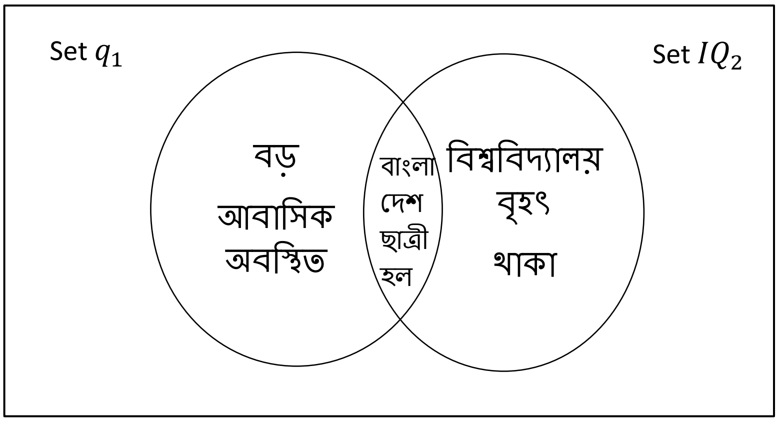
∴ Jaccard similarity is given by

$$sim(q_1, IQ_1) = \frac{q_1 \cap IQ_1}{q_1 \cup IQ_1}$$

$$= \frac{0}{9}$$

$$= 0$$

Again we will find the Jaccard similarity between set q_1 and set IQ_2.



From the Venn diagram, we have

$q_1 \cap IQ_2$ has 3 elements

$q_1 \cup IQ_2$ has 9 elements

∴ Jaccard similarity is given by

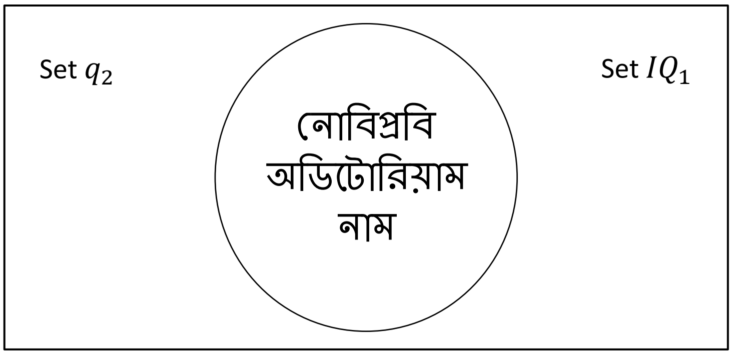
$$sim(q_1, IQ_2) = \frac{q_1 \cap IQ_2}{q_1 \cup IQ_2}$$

$$= \frac{3}{9}$$

$$= \frac{1}{3}$$

Since $sim(q_1, IQ_2) > sim(q_1, IQ_1)$, so the users question-1 can be found in IQ_2.

(ii) Now we will find the Jaccard similarity between set q_2 and set IQ_1.



From the Venn diagram, we have

$q_2 \cap IQ_1$ has 3 elements

$q_2 \cup IQ_1$ has 3 elements

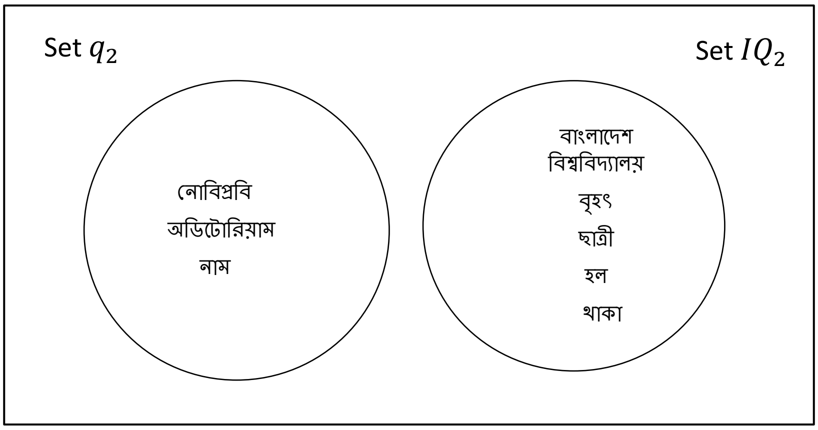
∴ Jaccard similarity is given by

$$sim(q_2, IQ_1) = \frac{q_2 \cap IQ_1}{q_2 \cup IQ_1}$$

$$= \frac{3}{3}$$

$$= 1$$

Similarly we will find the Jaccard similarity between set q_2 and set IQ_2.



From the Venn diagram, we have

$q_2 \cap IQ_2$ has 0 elements

$q_2 \cup IQ_2$ has 9 elements

∴ Jaccard similarity is given by

$$sim(q_2, IQ_2) = \frac{q_2 \cap IQ_2}{q_2 \cup IQ_2}$$

$$= \frac{0}{9}$$

$$= 0$$

Since $sim(q_2, IQ_1) > sim(q_2, IQ_2)$, so the users question-2 can be found in IQ_1.

Now the answer of user questions are

User Question-1: বাংলাদেশের সবচেয়ে বড় ছাত্রী হল কোথায় অবস্থিত?
BIIB Answer-1: নোবিপ্রবিতে অবস্থিত ।

User Question-2: এটির অডিটোরিয়ামের নাম কি?
BIIB Answer-2:বীর মুক্তিযোদ্ধা হাজী মোহাম্মদ ইদ্রিস অডিটোরিয়াম।

Chapter Eight

Experiment

Objective

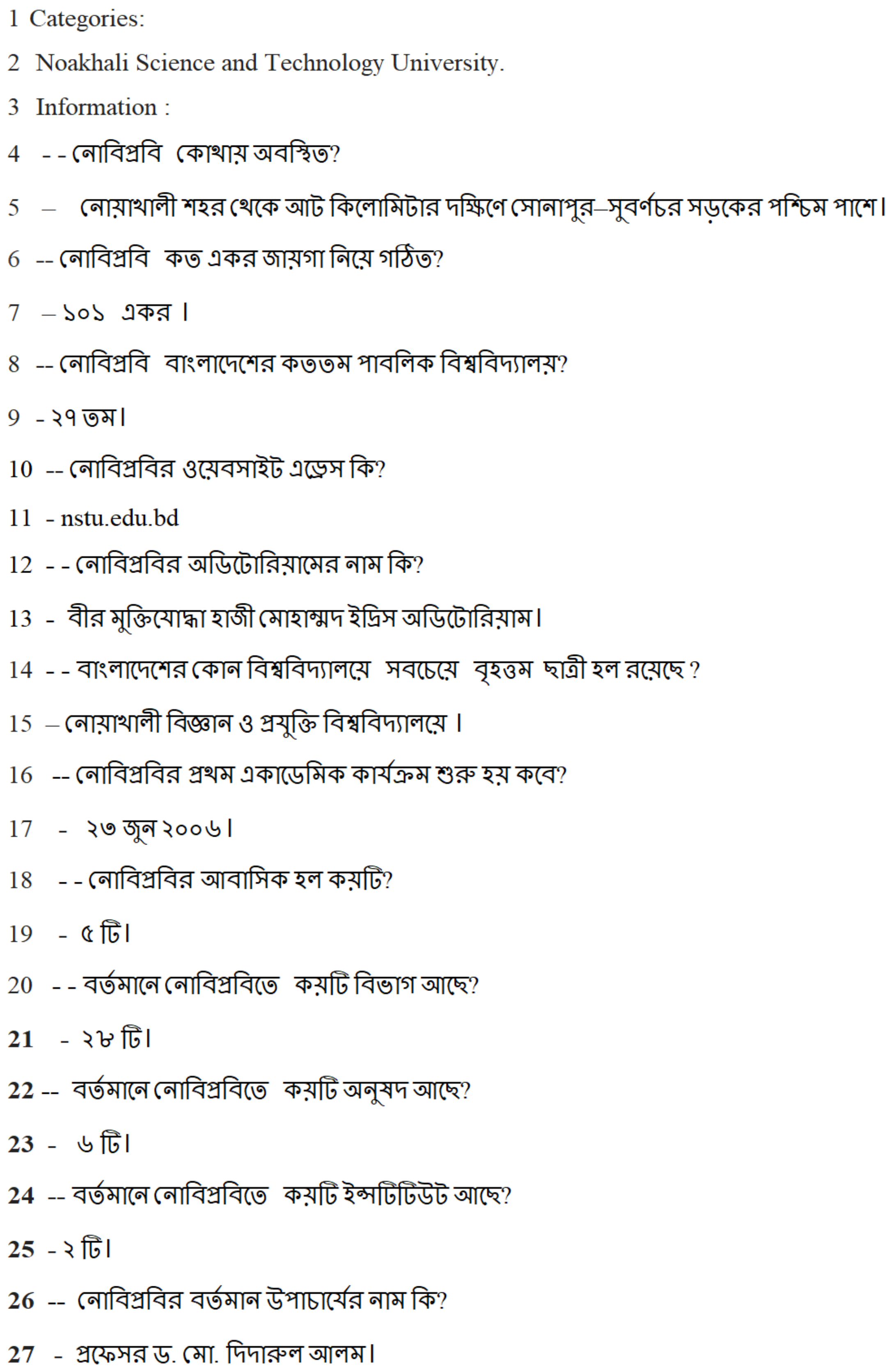
- To find experimental result of the previously discussed machine learning, mathematical & statistical procedures
- To set up AI tools like BLTK in order to train BIIB
- To obtain desired reply using several adapters

EXPERIMENTS

We describe a range of experiments to measure our proposed model the mathematical and statistical procedure for BIIB. In this section, first, we present the questions that we target to reply by the experiments and describe the experimental setup. Then, we discuss the performance and result of our propounded work.

8.1 Corpus

For the implication of BIIB we describe mainly five types of corpus. In the first corpus, there are 28,324 Bengali root words. The main aim of this corpus is to lemmatize Bengali words. The second one that contains 382 Bengali stop words is to remove the stop words from the inserted documents and questions. We have compiled 74 topics as informative documents like as hall information, department information, teacher information, library, NSTU nature, bus schedule etc. of Noakhali Science and Technology University (NSTU) that is our third corpus as questions with its relevant answer of document's information. In this work, we have originated 3127 questions from our inserted documents as our fourth corpus. Every question contains its corresponding answer. To test our data, we have created 2852 questions from relevant 74 topics. Every topic contains its most sentiment words for topic modeling.



```
Categories:
Noakhali Science and Technology University.
Information :
- - নোবিপ্রবি  কোথায় অবস্থিত?
–   নোয়াখালী শহর থেকে আট কিলোমিটার দক্ষিণে সোনাপুর–সুবর্ণচর সড়কের পশ্চিম পাশে।
-- নোবিপ্রবি  কত একর জায়গা নিয়ে গঠিত?
– ১০১  একর ।
-- নোবিপ্রবি  বাংলাদেশের কততম পাবলিক বিশ্ববিদ্যালয়?
- ২৭ তম।
-- নোবিপ্রবির ওয়েবসাইট এড্রেস কি?
- nstu.edu.bd
- - নোবিপ্রবির অডিটোরিয়ামের নাম কি?
-  বীর মুক্তিযোদ্ধা হাজী মোহাম্মদ ইদ্রিস অডিটোরিয়াম।
- - বাংলাদেশের কোন বিশ্ববিদ্যালয়ে  সবচেয়ে  বৃহত্তম  ছাত্রী হল রয়েছে ?
– নোয়াখালী বিজ্ঞান ও প্রযুক্তি বিশ্ববিদ্যালয়ে ।
-- নোবিপ্রবির প্রথম একাডেমিক কার্যক্রম শুরু হয় কবে?
-  ২৩ জুন ২০০৬।
- - নোবিপ্রবির আবাসিক হল কয়টি?
-  ৫ টি।
- - বর্তমানে নোবিপ্রবিতে  কয়টি বিভাগ আছে?
-  ২৮ টি।
--  বর্তমানে নোবিপ্রবিতে  কয়টি অনুষদ আছে?
-  ৬ টি।
-- বর্তমানে নোবিপ্রবিতে  কয়টি ইন্সটিটিউট আছে?
- ২ টি।
--  নোবিপ্রবির বর্তমান উপাচার্যের নাম কি?
-  প্রফেসর ড. মো. দিদারুল আলম।
```

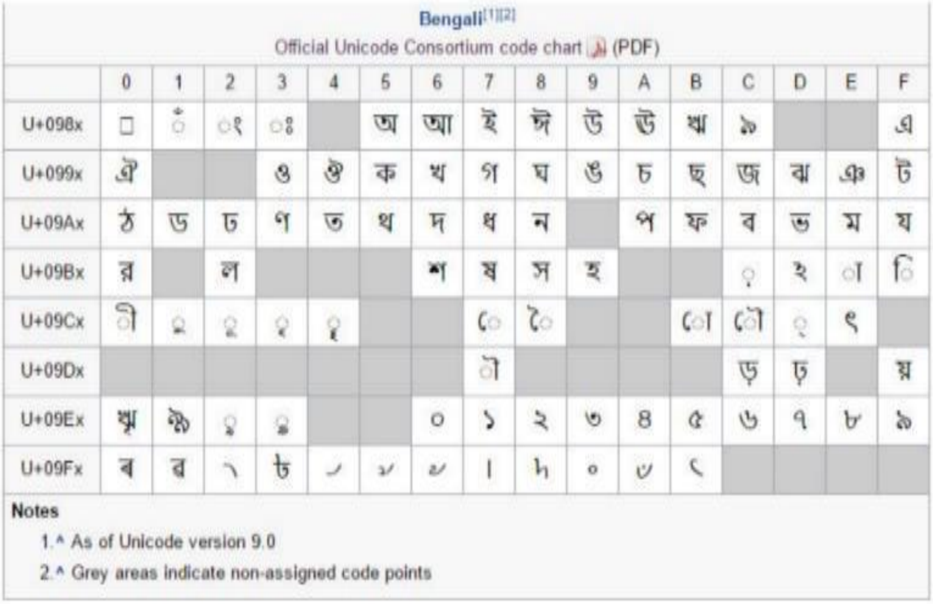
Fig.20: Data set of information

8.2 Experimental Setup

We have implemented both our propounded model and in Anaconda distribution with Python 3.7 programming language and executed them on a Windows 10 PC with an Intel Core i7, CPU (3.20GHz) and 8GB memory.

Python is a high-level object-oriented language (OOP) which is suitable for scientific examination and tools development. We used the Anaconda as the apportionment of Python. Anaconda creates the best stage for open source data science which is powered by Python.. For Natural Language Processing (NLP) we installed Natural Language Toolkit (NLTK). Python 3.7.0 is used for the implementation of Bengali chatbot ‘BIIB’ because of providing “Unicode Decode Error”. Unicode Decode Error is a runtime error which is caused by non-English language like Bengali with a large number of letters in the alphabet. The Unicode range of Bengali is 0980–09FF which has 11 vowels and 40 consonants. We use encoding="utf8" to encode the Bengali Language unlike English Language, it has consonant conjuncts, modifier, and other graphemes. So Bengali cannot be set up with ASCII decoding system. In many pre-processing steps, NLTK has no system for the Bengali Language such as to remove a stop word, lemmatization etc. In this case, we've created BLTK for the preprocessing all inserted questions and answer. For the sake of clearing data, removing stop words, we have constructed a tool for Bengali language processing mentioned as Bengali language Toolkit (BLTK) and also we have applied NLTK system in many pre-processing tasks. In BLTK tools, we developed SVD, LDA, TF-IDF and cosine similarity program based on Bengali Language.

In the following subsections, we recapitulate the experimental results that answer the above research questions. For the easy implication of code, we improve a python module of our model.



Bengali[1][2]
Official Unicode Consortium code chart (PDF)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+098x	□	◌ঁ	◌ং	◌ঃ		অ	আ	ই	ঈ	উ	ঊ	ঋ	ঌ			এ
U+099x	ঐ			ও	ঔ	ক	খ	গ	ঘ	ঙ	চ	ছ	জ	ঝ	ঞ	ট
U+09Ax	ঠ	ড	ঢ	ণ	ত	থ	দ	ধ	ন		প	ফ	ব	ভ	ম	য
U+09Bx	র		ল				শ	ষ	স	হ			◌়	ঽ	◌া	◌ি
U+09Cx	◌ী	◌ু	◌ূ	◌ৃ	◌ৄ			◌ে	◌ৈ			◌ো	◌ৌ	◌্	ৎ	
U+09Dx								◌ৗ					ড়	ঢ়		য়
U+09Ex	ৠ	ৡ	◌ৢ	◌ৣ			০	১	২	৩	৪	৫	৬	৭	৮	৯
U+09Fx	ৰ	ৱ	৲	৳	৴	৵	৶	৷	৸	৹	৺	৻				

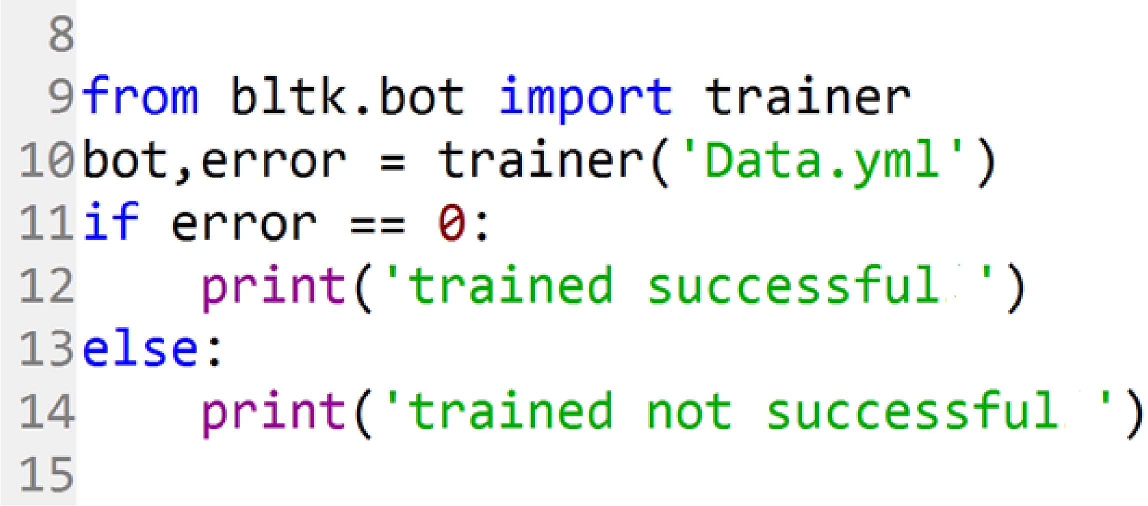
Notes
1. ^ As of Unicode version 9.0
2. ^ Grey areas indicate non-assigned code points

Fig. 21: Official Unicode Consortium Code Chart of Bengali.

8.3 Training BIIB

BLTK includes tools that help simplify the process of training a chat bot instance. BLTK's training process involves loading example dialog or data into the chat bot's database. This either creates or builds upon the graph data structure that represents the sets of known statements and responses. When a chat bot trainer is provided with a data set, it creates the necessary entries in the chat bot's knowledge graph so that the statement inputs and responses are correctly represented. The training technique of BLTK tool allows a Bengali chatbot to be trained using a list of strings where the list represents a conversation. In this case, the order of each response is based on its placement in a given conversation or the list of string.

For our implementation, we used the Bot Trainer Class of BLTK. At first, we have to create Bangla corpus in the data folder of the BLTK in the predefined format in JSON. So from the library, we set the trainer as training with the Bangla Corpus. We are providing the Code Snippet code of corpus trainer class based on the code for a better understanding of the trainer class.



```
from bltk.bot import trainer
bot,error = trainer('Data.yml')
if error == 0:
    print('trained successful ')
else:
    print('trained not successful ')
```

Fig 22: Code Snippet of the Bengali BLTK's Training example.

Before a user makes a question to BIIB, the bot will be trained. In a short way, the training sections are

1. Pre-processed whole informative questions corpus.
2. Every word is converted to numerical value according to its sentiment by TF-IDF model.
3. The dimension of the informative question corpus is reduced by the SVD.

8.4 Storage Adapters

The ‘bltk.bot’ comes with built-in adapter classes that allow it to connect to different types of databases. For our implementation, we will be using the Json File Storage Adapter which is a simple storage adapter that stores data in a JSON formatted file on the hard disk. This functionality makes this storage adapter very good for testing and debugging.

We will select the Json File Storage Adapter by specifying it in our chat bot’s constructor. The database parameter is used to specify the path to the database that the chat bot will use. The database.json file will be created automatically if it does not already exist.

8.5 Input Adapters

The ‘bltk.bot’s input adapters are designed to allow a chat bot to have a versatile method of receiving or retrieving input from a given source. It is required to add in parameters to specify the input and output terminal adapter. The input terminal adapter simply reads the user’s input from the terminal. The ‘bltk.bot’s input adapter class is an abstract class that represents the interface that all input adapters should implement. After getting input, the main job is the classify the text as a known or an unknown statement and pass it to the logic adapter after labeling the sentence as “known” or “unknown”. The goal of an input adapter is to get input from some source, and then to convert it into a format that ‘bltk.bot’ can understand. This format is the Statement object found in ‘bltk.bot’s conversation module. We used the variable input adapter for the implementation of BIIB. Variable input type adapter allows the chatbot to accept a number of different input types using the same adapter. This adapter accepts strings, dictionaries, and statements.

8.6 Topic Classification Adapter

The topic classification adapter is used to minimize the searching time complexity of a question or dialog. The topic classification adapter allows a chatbot to return the best topic related to a question. It is not necessary to find out all the relation with inserted information. The ‘bltk.bot’ helps to find out the relevant topic of a question then the logical adapter make the relation of its founded topic. For the topic classification, ‘bltk.bot’ uses the Latent Dirichlet Allocation (LDA) algorithm of Topic modeling.

8.7 Logic Adapters

Logic adapters determine the logic for how BIIB selects responses to a given input statement. It is possible to enter any number of logic adapters for a bot to use. If multiple adapters are used, then the bot will return the response with the highest calculated confidence value. If multiple adapters return the same confidence, then the adapter that is entered into the list first will take priority. The logic_adapters parameter is a list of logic adapters. In ‘bltk.bot’, a logic adapter is a class that takes an input statement and returns a response to that statement.

We employ Best Match Adapter for our chatbot. It is a logic adapter that returns a response based on known responses to the closest matches to the input statement. The Best Match logic adapter selects a response based on the best known match to a given statement. Once it finds the closest match to the input statement, it uses another function to select one of the known responses to that statement. The best match adapter uses cosine similarity function of TF-IDF value or Jaccard similarity to compare the input statement to known statements. Cosine Similarity compared two sentences based on dot product of two vectors. Jaccard Similarity makes the relation between user question and informative questions to reduce the size or dimension of a vector, 'bltk.bot' uses the Singular Value Decomposition (SVD) technique on the value of TF-IDF.

In the numerator, we count the number of items that are shared between the sets. In the denominator, we count the total number of items across both sets. Let us say we define sentences to be equivalent if 50% or more of their tokens are equivalent.

8.8 Response Selection Methods

Response selection methods determine which response should be used in the event that multiple responses are generated within a logic adapter. The 'bltk.bot' uses statement objects to hold information about things that can be said. An important part of how a chat bot selects a response is based on its ability to compare two statements to each other. This module contains various text comparison algorithms designed to compare one statement to another. We use the get first response method for the selection of a response. This method takes the input statement and selects the statement in the knowledge base which closely matches the input to the chatbot from a list of statement options to choose a response from.

8.9 Output Adapters

The output adapter allows the chatbot to return a response in as a Statement object. It is a generic class that can be overridden by a subclass to provide extended functionality, such as delivering a response to an API endpoint. Since our system is a text-based system we chose the "Text" format for our chatbot.

Chapter Nine

Results and Analysis

Objective

- To compare BIIB with different Chatbots
- To give a final analogy between cosine similarity and Jaccard similarity

RESULTS AND ANALYSIS

The difficulty of evaluation is intrinsic as each conversation is interactive and as generally the same conversation does not occur more than once; one slightly different answer will lead to a completely different conversation. Moreover, there is no clear sense of when such a conversation is “complete”. So for the evaluation, we have decided to compare our system with previous existing chatbots.

9.1 Comparison between English Chatbot and BIIB

Since there are no Chatbot in Bengali like BIIB, so there is any other Bengali Chatbot to compare with our work. Our work is at present the state of the art for the Bengali Intelligence Bot. So we compare BIIB with two English Chatbots which are Neural Conversational Machine (NCM) and Cleverbot. Neural Conversation Machine is a chatbot which can be trained end-to-end and this model chats by predicting the next sentence given the previous sentences or sentence with a few handcrafted rules. (Vinyals, 2015) So NCM is a generative based conversational agent whereas our work is a retrieval based system. We input some random questions to the NCM and our system for the comparative study between the two bot. Science our Bot is informative and developed with the data set of Noakhali Science and Technolgoy University, so the data set isn’t available for Cleverbot and NCM. That’s why we created a basic dataset and added to corpus with the topic of ‘Basic Conversation’

Users	Cleverbot	BIIB
What is machine learning? মেশিন লার্নিং কি ?	Hang on, my brother wants to talk to you.	মেশিন লার্নিং আর্টিফিশিয়াল ইন্টেলিজেন্সের একটা বিভাগ যেখানে ইন্টেলিজেন্ট সিস্টেম তৈরি করা হয় ডেটাসেট কিংবা ইন্টারঅ্যাক্টিভ এক্সপেরিয়েন্সের মাধ্যমে।
What is python? পাইথন কি ?	A programming language.	পাইথন একটি অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ল্যাংজুয়েজ।
What is Artificial Intelligence? কৃত্রিম বুদ্ধিমত্তা কি?	A computer program that simulates the thoughts of a real human being, therefore making it seem as though it has intelligence when it actually does not. It has a series of pre-programmed responses	মানুষের বুদ্ধিমত্তা ও চিন্তা শক্তিকে কৃত্রিম উপায়ে প্রযুক্তিনির্ভর করে যন্ত্রের মাধ্যমে বাস্তবায়ন করাকে কৃত্রিম বুদ্ধিমত্তা বলে।
Which algorithms are used to build you? তোমাকে তৈরি করতে কোন কোন এলগরিদম ব্যবহার করা হয়েছে ?	Maybe I have a human brain; but, is it possible?	আমাকে ডেভেলপ করার জন্য কিছু ম্যাথমেটিক্যাল ও স্ট্যাটিস্টিক্যাল এলগরিদম ব্যবহার করা হয়েছে।

Table 7: Comparison between CleverBot and BIIB.

Users	Mitsuku	BIIB
What’s your mobile number? তোমার মোবাইল নাম্বার কি?	That information is confidential.	আমার কোন মোবাইল নেই।আমার সাথে যোগাযোগ করতে আমার প্রোগ্রামটি রান করুন।
How old are you? তোমার বয়স কত?	I am 18 years old.	আমি এখনও তরুন ।
What is your address? তোমার ঠিকানা কি?	I am in Leeds.	আমার প্রোগ্রামটি যে কম্পিউটারে ইনস্টল থাকবে সেটিই আমার ঠিকানা ।
Have you feeling? তোমার অনুভূতি আছে?	No I don’t think I have any feel. But I do have a collection of gossipe.	অনুভূতি থাকে মানুষের। আমি একটি কৃত্রিম বুদ্ধিমত্তা প্রোগ্রাম আমার কোন অনুভূতি নাই।

Table 8: Comparison between Mitsuku and BIIB.

9.2 Final Result

To test our proposed BIIB, We have created 2852 questions as testing data from selected 74 topics of Noakhali Science and Technology University (NSTU), we obtained 96.22 % accuracy in cosine Similarity and 82.64% in Jaccard similarity. In cosine similarity, the number of the correct answers is 2744 and incorrect is 108. In Jaccard similarity, the number of the correct answers is 2356 and incorrect is 495. The performance to reply a question is better and flexible with the lowest time complexity and instant time in cosine similarity.

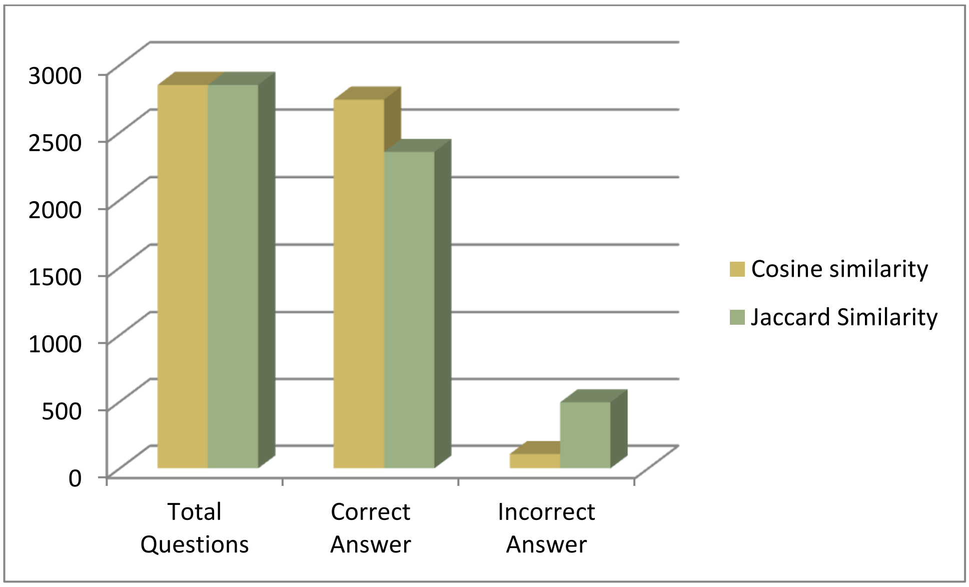


Fig. 23: Comparison between cosine similarity and Jaccard similarity.

Chapter Ten

Conclusion and Future Work

Objective

- To give the gist of whole project work
- To share ideas for the future development of BIIB

CONCLUSION AND FUTURE WORK

The main challenge of this project paper is to implement a Bengali intelligence bot for information retrieval. We have showed the theoretical and experimental methodology of our proposed work. In this scientific paper, we have described two procedures using machine learning, mathematics and statistics. To establish the full methodology, we have followed some procedures like as pre-processing, time and space reduction, and established relation between information and questions. Different types of algorithms and methods are used such as lemmatization, anaphora resolution, SVD, NMF, TF-IDF, cosine similarity and Jaccard similarity. We have used matrix factorization process to reduce time complexity and space reduction. The all perform actions have been processed with the Bengali Language as part of BNLP. We have tested our proposed BIIB, spectacled the accuracy, correct and incorrect results and comparison between two procedures.

In the future, the proposed BIIB system can be enabled for the purpose of educations, industry, business and personal tasks. An advance BIIB can be shaped with the assist of Deep Learning algorithms such as Recurrent Neural Network (RNN) by processing the BNLP.

For future work, we plan to train a chatbot with a neural network model provided with the corpus we have got from this project. Also, we can try a crowd-sourced model to enrich the database of the chatbot by integrating this chatbot in a website. Our proposed context can be incorporated in future work with the above mentioned two algorithms.

To produce sensible responses, systems may need to incorporate with both linguistic context and visual context. During or after long conversations, people can recall what has been said and what information has been exchanged. Other kinds of contextual data such as date/time, location or information about a user can be added in future. It requires a large collection of conversation corpus. To make it possible, a generative open domain system can be incorporated. As we do not have that much data to make it generative, this can be done in future.

We can optimize the automated system by making the chatbot a voice-enabled system to reply in pictorial representation for understanding people better with low literacy.

REFERENCES

[1] B. A. Shawar, and E. Atwell, “Different measurements metrics to evaluate a chatbot system,” Proceedings of the Workshop on Bridging the Gap: Academic and Industrial Research in Dialog Technologies, 2007.

[2] J.Weizenbaum, “ELIZA — A Computer Program For the Study of Natural Language Communication Between Man And Machine,” Commun. ACM, vol. 9, no. 1, pp. 36–45, 2016.

[3] B. A. Shawar and E. Atwell, “A comparison between ALICE and Elizabeth chatbot systems,” Raport instytutowy, University of Leeds, 2002.

[4] C. Yin and Z. Jinran, “Seniors prove WeChat is not just for young,” CHINA CHANNEL, 2017.

[5] Antol, Stanislaw, et al. "Vqa: Visual question answering." *Proceedings of the IEEE international conference on computer vision*. 2015.

[6] Yang, Yi, Wen-tau Yih, and Christopher Meek. "Wikiqa: A challenge dataset for open-domain question answering." *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. 2015.

[7] In Bio-medical QAS question types including factoid, list based, summary and yes/no type questions that generate both exact and wellformed ‘ideal’ answers,IEEE,2017.

[8] Huang, Jizhou, Ming Zhou, and Dan Yang. "Extracting Chatbot Knowledge from Online Discussion Forums." *IJCAI*. Vol. 7. 2007.

[9] Shawar, Bayan Abu, and Eric Atwell. "Chatbots: are they really useful?." *Ldv forum*. Vol. 22. No. 1. 2007.

[10] Ghose, Supratip, and Jagat Joyti Barua. "Toward the implementation of a topic specific dialogue based natural language chatbot as an undergraduate advisor." *2013 International Conference on Informatics, Electronics and Vision (ICIEV)*. IEEE, 2013.

[11] Ranoliya, Bhavika R., Nidhi Raghuwanshi, and Sanjay Singh. "Chatbot for university related FAQs." *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2017.

[12] Ranoliya, Bhavika R., Nidhi Raghuwanshi, and Sanjay Singh. "Chatbot for university related FAQs." *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2017.

[13] Jenkins, Marie-Claire. *Designing Service-Oriented Chatbot Systems Using a Construction Grammar-Driven Natural Language Generation System*. Diss. University of East Anglia, 2011.

[14] Bickmore, T., Schulman, D.: Practical approaches to comforting users with relational agents. In: ACM SIGCHI Conference on Human Factors in Computing Systems (2007)

[15] Elton, Daniel C., et al. "Using natural language processing techniques to extract information on the properties and functionalities of energetic materials from large text corpora." *arXiv preprint arXiv:1903.00415* ,2019.

[16] Champa, H. N. "An Extensive Review on Sensing as a Service Paradigm in IoT: Architecture, Research Challenges, Lessons Learned and Future Directions." *International Journal of Applied Engineering Research* 14.6 ,1220-1243, 2019.

[17] Chaves, Ana Paula, and Marco Aurelio Gerosa. "How should my chatbot interact? A survey on human-chatbot interaction design." *arXiv preprint arXiv:1904.02743* ,2019.

[18] Chaves, Ana Paula, and Marco Aurelio Gerosa. "How should my chatbot interact? A survey on human-chatbot interaction design." *arXiv preprint arXiv:1904.02743* ,2019.

[19] Lee, Paul SN. "A Study of Startups in Hong Kong." ,2018.

[20] Zhu, Xiaoming. "Introduction: From the Industrial Economy to the Digital Economy: A Giant Leap—Research on the “1+ 10” Framework of the Digital Economy." *Emerging Champions in the Digital Economy*. Springer, Singapore, 1-65, 2019.

[21] McAfee, R. Preston. "Review of AI Superpowers: China, Silicon Valley and the New World Order, by Kai-Fu Lee." 1-6, 2019.

[22] Morris, Margaret E. *Left to Our Own Devices: Outsmarting Smart Technology to Reclaim Our Relationships, Health, and Focus*. MIT Press, 2018.

[23] Li, Rita Yi Man. "Software engineering and reducing construction fatalities: An example of the use of Chatbot." *An Economic Analysis on Automated Construction Safety*. Springer, Singapore, 105-116, 2018.

[24] Borschev, Vladimir, and Barbara H. Partee. "Andrei Anatolievich Zalizniak In Memoriam." *Journal of Slavic Linguistics* 26.1 ,3-16, 2018.

[25] Dereza, Oksana. "Lemmatization for Ancient Languages: Rules or Neural Networks?." *Conference on Artificial Intelligence and Natural Language*. Springer, Cham, 2018.

[26] Dereza, Oksana. "Lemmatization for Ancient Languages: Rules or Neural Networks?." *Conference on Artificial Intelligence and Natural Language*. Springer, Cham, 2018.

[27] Song, B., Zhuo, Y., & Li, X. (2018, June). Research on Question-Answering System Based on Deep Learning. In *International Conference on Sensing and Imaging* (pp. 522-529). Springer, Cham.

[28] Zhang, W.: Chinese Question Answering System Technology and Application. Electronic Industry Press, Beijing (2016)

[29] Song, Bo, Yue Zhuo, and Xiaomei Li. "Research on Question-Answering System Based on Deep Learning." *International Conference on Sensing and Imaging*. Springer, Cham, 2018.

[30] Song, Bo, Yue Zhuo, and Xiaomei Li. "Research on Question-Answering System Based on Deep Learning." *International Conference on Sensing and Imaging*. Springer, Cham, 2018.

[31] P. Galvin, “Alibaba Invests in AI Startup,” Tech Exec., 2016.

[32] R. Dillet, “Sinovation Ventures’ Dr. Kai-Fu Lee is betting big on artificial intelligence,” TechCrunch, 2016.

[33] Roy, Monjoy Kumar, et al. "Suffix Based Automated Parts of Speech Tagging for Bangla Language." *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*. IEEE, 2019.

[34] Gupta, V., Joshi, N., & Mathur, I. Advanced Machine Learning Techniques in Natural Language Processing for Indian Languages. In *Smart Techniques for a Smarter Planet*(pp. 117-144). Springer, Cham, 2019.

[35] Wang, K., Wang, H., Wang, Z., Yin, Y., Mao, L., & Zhang, Y. Method for pigment spectral matching identification based on adaptive levenshtein distance. *Optik*, *178*, 74-82, 2019.

[36] Shi, Jianlin, and John F. Hurdle. "Trie-based rule processing for clinical NLP: A use-case study of n-trie, making the ConText algorithm more efficient and scalable." *Journal of biomedical informatics* 85 ,106-113, 2018.

[37] Shi, Jianlin, and John F. Hurdle. "Trie-based rule processing for clinical NLP: A use-case study of n-trie, making the ConText algorithm more efficient and scalable." *Journal of biomedical informatics* 85 ,106-113, 2018.

[38] Jelodar, Hamed, et al. "Latent Dirichlet Allocation (LDA) and Topic modeling: models, applications, a survey." *Multimedia Tools and Applications* (2018): 1-43,2017.

[39] Chou, Yi-Chi, Chun-Yen Chao, and Han-Yen Yu. "A Résumé Evaluation System Based on Text Mining." *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE, 2019.

[40] Young, T., Hazarika, D., Poria, S., & Cambria, E. Recent trends in deep learning based natural language processing. *IEEE Computational intelligenCe magazine*, *13*(3), 55-75, 2018.

[41] Lesnikowski, Alexandra, et al. "Frontiers in data analytics for adaptation research: Topic modeling." *Wiley Interdisciplinary Reviews: Climate Change* (2019): e576,2017.

[42] Song, Yicheng, Nachiketa Sahoo, and Elie Ofek. "When and How to Diversify—A Multicategory Utility Model for Personalized Content Recommendation." *Management Science* ,2019.

[43] Mizoguchi, Takehiko, and Isao Yamada. "Hypercomplex tensor completion with Cayley-Dickson singular value decomposition." *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018.

[44] Kim, Heejong, Joseph Piven, and Guido Gerig. "Longitudinal structural connectivity in the developing brain with projective non-negative matrix factorization." *Medical Imaging 2019: Image Processing*. International Society for Optics and Photonics, Vol. 10949,2019.

[45] Caron, Mathilde, et al. "Deep clustering for unsupervised learning of visual features." *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018.

[46] Levandoski, Andrew, and Jonathan Lobo. "Document and Topic Models: pLSA and LDA." 2018.

[47] Luo, Tingjin, et al. "Dimension reduction for non-Gaussian data by adaptive discriminative analysis." *IEEE transactions on cybernetics* 99 ,1-14. 2018.

[48] Mustafi, D., and G. Sahoo. "A hybrid approach using genetic algorithm and the differential evolution heuristic for enhanced initialization of the k-means algorithm with applications in text clustering." *Soft Computing* ,1-18. 2018.

[49] Schneider, Tizian, Nikolai Helwig, and Andreas Schütze. "Automatic feature extraction and selection for condition monitoring and related datasets." *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. IEEE, 2018.

[50] Papi, M. and Caracciolo, G.Principal component analysis of personalized biomolecular corona data for early disease detection. *Nano Today*, *21*, pp.14-17. 2018.

[51] Caudle, Kyle, Randy Hoover, and Karen Braman. "Multilinear Discriminant Analysis Through Tensor-Tensor Eigendecomposition." *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2018.

[52] Ireland, Robert, and Ang Liu. "Application of data analytics for product design: Sentiment analysis of online product reviews." *CIRP Journal of Manufacturing Science and Technology* 23 ,128-144. 2018.

Common Abbreviations with Definitions

AI (Artificial Intelligence): *Artificial intelligence (AI) is an area of computer science that emphasizes the creation of intelligent machines that work and reacts like humans.*

Anaphora Resolution: *An anaphora is a rhetorical device in which a word or expression is repeated at the beginning of a number of sentences, clauses, or phrases.*

BIIB (Bengali Informative Intelligence Bot): *BIIB is our Proposed Work.*

BLTK (Bengali Language Toolkit): *The BLTK is a Python package for natural language processing.*

BNLP (Bengali Natural Language Processing): *BNLP is the ability of a computer program to understand the bengali language as it is spoken.*

BRE (Bengali Regular Expression): *BRE are patterns used to match character combinations in strings.*

Corpus: *Corpus refers dataset.*

Cosine Distance: *Cosine distance is a* measure of dissimilarity *between two non-zero vectors of an* inner product space *that measures the* cosine *of the angle between them.*

Cosine Similarity: *Cosine similarity is a* measure of similarity *between two non-zero vectors of an* inner product space *that measures the* cosine *of the angle between them.*

Chatbot: *A chatbot is artificial intelligence (AI) software that can simulate a conversation with a user.*

Cleverbot: *Cleverbot is an English chatbot which was created by British AI scientist Rollo Carpenter.*

DBSRA (Dictionary Based Search by Removing Affix): *DBSRA is a lemmatization techniques specially designed for Bengali Language developed by Shanto Sutra Dhar et al.*

Deep learning: *Deep learning is an artificial intelligence function that imitates the workings of the human brain in processing data and creating patterns for use in decision making.*

Dimension Reduction: *Dimension Reduction is a linear algebraic technique to reduce the vector space.*

Gradient Descent: *Gradient descent algorithm is an iterative process that takes us to the minimum of a function.*

Lemmatization: *Lemmatization is the process of converting a word to its base form.*

Least Square method: *Least square method is a statistical method used to determine a line of best fit by minimizing the sum of squares.*

LDA (Latent Dirichlet Allocation): *LDA is a topic modeling using probabilistic techniques.*

LSA (Latent Semantic Analysis): *LSA is a topic modeling using matrix factorization techniques.*

LSI (Latent Semantic Indexing): *LSI is a topic modeling using matrix factorization techniques.*

Jaccard Distance: *Jaccard similarity is a statistic for comparing the dissimilarity and diversity of sample sets which are distant. It also called indexing.*

Jaccard Similarity: *Jaccard similarity is a statistic for comparing the similarity and diversity of sample sets which are distant. It also called indexing.*

Levenshtein Distance: *Levenshtein Distance or Edit distance is a procedure where dynamic programming technique is used to measure the relationship between two strings.*

Matrix factorization: *A matrix decomposition or matrix factorization is a factorization of a matrix into a product of matrices.*

ML (Machine Learning): *Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.*

Mitsuku: *Mitsuku is a chatbot created from AIML technology by Steve Worswick.*

NLP (Natural language processing): *NLP is the ability of a computer program to understand human language as it is spoken.*

NLU (Natural Language Understanding): *NLU is Artificial Intelligence that uses computer software to interpret text and any type of unstructured data.*

NLTK (Natural Language Toolkit): *The Natural Language Toolkit (NLTK) is a Python package for natural language processing.*

NMF (Nonnegative Matrix Factorization): *NMF is a matrix factorization method where we constrain the matrices to be nonnegative.*

NSTU: *Noakhali Science and Technology University.*

PCA (Principal component analysis): *Principal component analysis is an unsupervised algorithm that creates linear combinations of the original features.*

RE (Regular Expression): *Regular expressions are patterns used to match character combinations in strings.*

RNN (Recurrent Neural Network): *RNN is a type of Neural Network where the output from previous step is fed as input to the current step.*

Stop words: *Stop words refer the words that does not influence on documents or sentences.*

SVD (Singular Value Decomposition): *SVD is a matrix decomposition method for reducing a matrix to its constituent parts in order to make certain subsequent matrix calculations simpler.*

Space complexity: *Space complexity is a measure of the amount of working storage an algorithm needs.*

TF-IDF (Term Frequency-Inverse Document Frequency): *TF-IDF is an information retrieval technique that weighs a term's frequency (TF) and its inverse document frequency (IDF).*

Trie: *Trie or Prefix Tree is a tree-based data structure and exoteric for the mechanisms to store data.*

Topic Modeling: *Topic modeling a topic model is a type of* statistical model *for discovering hidden sentiment.*

Topic Classification: *Topic classification is a method to classify similar test into a group.*

Time Complexity: *Time complexity is a function describing the amount of time an algorithm takes in terms of the amount of input to the algorithm.*