International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-9 Issue-3, February 2020

Enhancing the Performance of Semantic Search in Bengali using Neural Net and other Classification Techniques

Arijit Das, Diganta Saha

Abstract: ***To know the information from the internet searching is one of the most important part for any user. In case of 'Syntactic Search' keyword based matching technique is used. Search accuracy is improved applying the filter like location, preference, user-history etc. However, it can happen that the user query or question and the best available answer or result in the internet domain has no terms in common or ignorable number of terms is common. In such case syntactic search cannot give the desired output. The role of 'Semantic Search' becomes prevalent in this scenario. The execution of semantic search faces challenge due to unavailability of resources like WordNet, Ontology, Annotation etc. An end to end algorithm is described to improve the accuracy of the semantic search in this work. Four classification techniques are used. They are ANN, Decision Tree, SVM and Naïve Bayes. Dataset is provided from the TDIL project of the Ministry of Electronics and IT, Govt. of India. The repository contains 86 categories of text having more than a million sentences. After getting the impressive result for the Bengali language test run was done for other Indian languages and a very good result is achieved. This research is extremely useful for the automatic question answering system, semantic similarity analysis, e-governance and m- governance.***

Keywords: Semantics, ANN, SVM, Naive Bayes, Decision Tree, Classification Techniques, Semantic Search.

I. INTRODUCTION

Now a day, we are surrounded by the digital content. We use internet for any information. Searching is the most common method to get any information from the net. If we see in to the linguistic map of the world, we will find that majority numbers of countries in the world are multilingual. The citizens of these countries have various mother tongues. They use different languages for reading, writing and speaking. In case of India, we have 22 scheduled languages which are being used by the Governments in the states and Central Government for day to day office work. Except that there are at least 2000 languages which are being used in various corners of the country. In rural India there is a lack of knowledge of English and in South India there is lack of use of Hindi. In this context the linguists have found that major language is changed in every 80 km and dialect of the language changes in every 8 km. As a whole, India has tremendous language diversity. This diversity is also very common in other parts of the world as well. Human being is most comfortable to communicate in their mother tongue. Psychologists have found that we always think or dream in our mother tongue. So, for comfort or for the lack of knowledge of other languages users always prefer to use their mother tongue. This is also very much true for internet users as well.

Google has already achieved a lot of success in online translation. Google transliterate is widely used for translation of any phrase from one language to another language. Here we have worked in a different problem or gap which is not solvable by language translation.

For the first case, a query or question is fired in language A and the answer to that or information related to that is available in language B. Search Engine is unaware about that answer as it's never possible to translate the query in to thousands of different languages. In the second case, the query and the answer are available in the same language but there is no word in common. In both the cases syntactic search fails due to the unavailability or less availability of the common terms or common words between the query and the answer.

Semantic Search tries to find the meaning of the query and retrieves the result accordingly. Over the time researchers have tried to implement the semantic search I different ways. In case of popular languages where a lot of resources like WordNet, Ontology etc. are available it is giving a good result as well. But in case of Bengali or other Indian languages where WordNet is not complete or ontology is still not developed a different approach is required. We have used classification technique in this work to map a query and its result contextually.

Same word has different meaning in different context. "I have a antique table." Here the word 'Table' is an object and used as a noun. "The parliamentary committee will table the proposal today." Here the word 'Table' is verb and it means 'to present'. The most popular 'Lesk Algorithm' is used for Word Sense Disambiguation (WSD). The algorithm suggests to use the nearby words and lookup them in the dictionary and WordNet for determining a particular meaning for a word among different meanings.

Here the challenge, to understand the meaning is not of a word rather of the whole query. Machine Learning techniques are used to extract the meaning of query and retrieve the contextually related answer. Machine learning are generally three types- supervised, unsupervised and semi supervised. In case of supervised learning a specific rule sets are mentioned. Building this rule or training set is time consuming and laborious job.

Revised Manuscript Received on February 05, 2020.
* Correspondence Author
Arijit Das, Department of Computer Science and Engineering, Faculty of Engineering and Technology, Jadavpur University, Kolkata, West Bengal.
Diganta Saha, Professor of the Department of Computer Science and Engineering in Jadavpur University.

In case of unsupervised learning the system learns using some statistical or mathematical formulas. Accuracy increases with more iterations of run and larger training set. Initial accuracy is generally much lower in case of unsupervised learning in comparison to supervised learning. In case of semi-supervised learning a hybrid approach is used to tradeoff between the labour and the initial accuracy.

In this work a collection of novels and texts written or translated in Bengali is used. This repository contains 86 different categories of text marked with text category and the sentence header. Total size of the repository is 6 GB with only textual data containing millions of sentences. The repository is developed as a part of Technology Development of Indian Language (TDIL) project of the MeitY, of Govt. of India.

After design and development of the algorithm and processing of the corpus with the algorithm a set of questions are fired and answers are collected. Final result and the accuracy are evaluated by the experts.

The questions are prepared and collected from the native speakers arbitrarily or without any biasness. It is quite common that the question and the answer sentence have no terms in common. The examples of the sentences are

"Ram er Ma er Nam ki?" (What is the mother's name of Ram?)

"Ronaldo Kon Club e khelen? " (In which club Ronaldo plays?)

A set of 250 questions are fired and 97.6 percent accuracy have been achieved.

In the next sections, previous work is described in the 'Related Work' section. The logical definition of the problems and sub-problems to be solved is given in the "Problem Statement" section. Our approach to solve the problem is provided in the "Proposed Approach" section. The algorithm is provided in the "Methodology" section. In the next few sections result, the challenges faced and the future scope of the improvement is given.

II. RELATED WORK

In general, WordNet and Ontology are used to resolve the query semantically. Problem occurs when the said resources are not available or under developed. Researchers have used machine learning, deep learning, data mining, knowledge graph, vectors and other techniques to challenge the semantic similarity measurement.

In recent research [1, 2] scientists have used graph based methods to measure the semantic similarity of two sentences or two phrases. A sentence is parsed to a tree or graph and the graph similarity algorithm is used to find out the matching index of two graphs. [3] Proposed to use the Euclidean distance to find out the similarity between two graphs.

In case of mixed language [6] used deep neural network. Code mixed language which is already classified or tagged was used as training set. With the iterations deep neural network learned the relation between sentences represented in mixed language and gave the output.

Dong and Hussain [7] have used ranking methodology to determine the appropriate answer for a service request for digital transportation system. The initial ranking was presented to user and based on the feedback the system learned to improve.

Portman [8] has used the fuzzy algorithm to the ontology to determine the reputation score for a query-answer pair. Dong [9] proposed his developed framework with the fuzzy backbone to determine the service request for digital transportation system.

Chandu and his group [10] proposed web mining to answer the fact-based questioning. They have proposed own framework to implement the same. Web mining and text mining are used vividly to answer the query.

A match-based algorithm is proposed in [11]. They have used WordNet to resolve any query semantically. Match based algorithm are successful in case of high resource language where the WordNet is fully developed.

LitVar a semantic search engine is proposed by Allott et al [12]. The search engine is used to search the genome related data in the database. NLP is used to resolve various bio-informatics based query. Sahani and his group [13] has proposed different Word Sense Disambiguation based techniques to measure the similarity between words. Word to Vector algorithm is used to find the similarity in the ontological classes [14].

Zhu and his colleagues [15] have proposed an asymmetric measurement technique to match the job query and the suggested jobs. Semantic measurement of similarity ensures that not only the matching word based jobs are retrieved rather based on qualification, experience and searching prospective jobs are shown.

H. Bast [16] has proposed mining-based methodology in the text database to form the knowledge graph and sort out the similarity.

III. PROBLEM STATEMENT

The problems which have been attacked and resolved by this work is described in structured form–

a. Taking the query as input and process it.
b. Determine the category or class of the query.
c. Map the major class or major category of text where the answer of the query is available.
d. Drill down to the exact portion of the specific class of text where the answer is actually present.
e. If more than one answer are retrieved, how to rank them or remove the ambiguity.
f. Extraction and formation of answer and give it as output.

IV. PROPOSED APPROACH

To retrieve the answer for any query in the semantic way classification techniques are proposed in this work. The repository is already categorized into 86 different classes of text. Four classification techniques namely ANN, Naïve Bayes, SVM and Decision Tree are used as classification techniques. Using each of these algorithms and training set, a model is prepared. In the next phase the query and the model are passed and the class of the query (among one of the 86 classes) is determined. Four classification techniques

are used to remove any algorithmic biasness and make the result more perfect.

A series of preprocessing works are required to be applied to both of the query and the sentences in the repository as well. Similarization of fonts, removal of uneven spaces, collection of punctuation marks, POS tagging of the words present in the sentence, identification Function Word and Content Word, Named Entity Recognition, Root Verb extraction, identification of person-number-tense-gender from the morphed verb and pronoun, lemmatization etc. are done as part of preprocessing.

After initial classification of the query, the specific category of the text needs to be processed, to locate the part of text where the exact answer is hidden. For that operation, recursively four classification methods are applied using the attributes like tense, number (singular or plural), person, gender (male or female), named entity, sense etc. to drill down to the portion of the text where prospective response of the query exists. Atomicity level up to the sentence is achieved in this process.

Ambiguity arises when the predicted class is different by different algorithms. Weighted average technique is used to resolve such ambiguity. A weight of 0.25 is distributed to all the algorithms. When any specific predicted class receives more than 0.50 weight, that class is chosen.

The brief theory of the algorithms used are given below.

A. DECISION TREE

The classification technique "Decision Tree" forms a tree structure of logical expressions from the training data. First different attributes are selected and then based on the 'True' and 'False' value of those attributes branch or path to go to the next level is chosen. In the next level based on the value of the attribute right or left path is chosen.

The concept of the decision tree gets cleared from an example. The model generated by the decision tree from a training dataset is shown in the Figure 4.1. This model is to predict the probability of survival for a passenger in the Titanic. If the passenger is female her survival probability is 36%. If the passenger is male and age is less than 9.5 then chance of his death is 61%. If he is older than 9.5 years age and his spouse and siblings are more than 2.5 in counting then his survival chance is 2%. This model is prepared from the training data set using decision tree algorithm. Here sex, age, count of sibling and spouse are the attributes based on which decision is taken. After preparing the model, when a new testcase comes, its probability of survival can be predicted based on the value of those three attributes. This is the way how decision tree works.

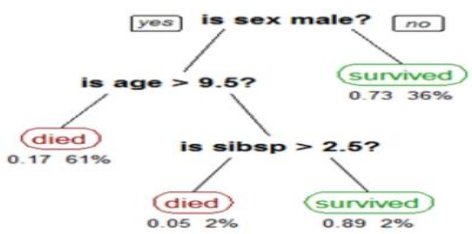

Figure 1. Decision Tree

B. NAIVE BAYES

Naïve Bayes is the classification technique which works on the Bayesian probabilistic theory. Bayes theorem tells that,

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Where P(A|B) is the conditional probability which is defined as likelihood of event A occurring given that B is true.
P(B|A) is the conditional probability which is defined as likelihood of event B occurring given that A is true.
P(A) is the likelihood of event A occurring.
P(B) is the likelihood of event B occurring.

For an example, suppose the document containing the words "Actress became Education-Minister" needs to be classified. It may be included in the class 'Films' or 'Politics'. Naïve Bayes will classify the document using the formula stated above. Here [P(Films) * P("Actress"|Films) * P("became"|Films) * P("Education-Minister"| Films)] is compared with [P(Politics) * P("Actress"|Politics) * P("became"|Politics) * P("Education-Minister"| Politics)]. The Naïve Bayes classifier classifies the document based on the higher value of the mentioned comparison.

C. SUPPORT VECTOR MACHINE

SVM or Support Vector Machine is a supervised classification technique where a set of hyperplanes are used in a multi-dimensional space to segregate or classify the data. The data is represented as a point in the n dimensional space based on its value. It is common, that the data points are distributed in such a fashion that linear classifier is unable to classify them. In such case, the data is transformed to some higher dimensional space and hyperplane is used to separate them.

SVM tries to find out hyper plane with maximum margin between sets of objects. In the Figure 2.a. possible hyperplanes are shown with the green lines. In the Figure 2.b. optimal hyperplane is shown. The hyperplane ensures the maximum margin. Red square and blue rounds are separated by the optimal hyperplane. Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane.

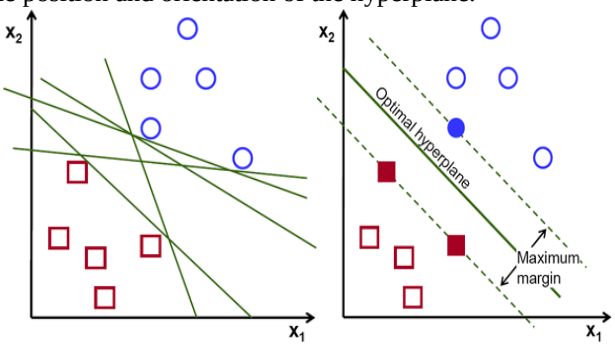

Figure 2.a. Possible Hyperplanes **Figure 2.b. Optimal Hyperplane**
(Courtesy: Text Mining Book of Han Kamber)

D. ARTIFICIAL NEURAL NETWORK

ANN or Artificial Neural Network is conceptualized from the learning pattern of the human brain. A basic ANN is given in the Figure 4.2.

There is an input layer which takes input, an output layer which gives the final output and a set of hidden layers which are involved in the processing, decision making tasks etc. A single node called neuron is the basic block of the ANN. It receives input, combine the input with its internal state (which is called activation) and athreshold(which is optional)using an 'activation function' and produce output using an 'output function'.

The structure showed in the Figure 3. is of the basic ANN. In the figure w[x,y] means x is the number of inputs to the each of the neuron and y is the number of outputs from each neuron in the layer. W is the weight which is used to increase or reduce the importance of any input by multiplying it with a factor. Based on the architecture, ANN can be of different types like feed forward (the simplest one shown in the Figure 4.2.), feedback neural network, convolutional neural network, recurrent neural network etc.

The ANN gains its popularity from the year 2010 onwards. It increased the efficiency of signature detection, image recognition up to 15 percent compared to other methods. The challenges of the ANN are long processing time, requirement of huge training data, necessity of high performance hardware etc.

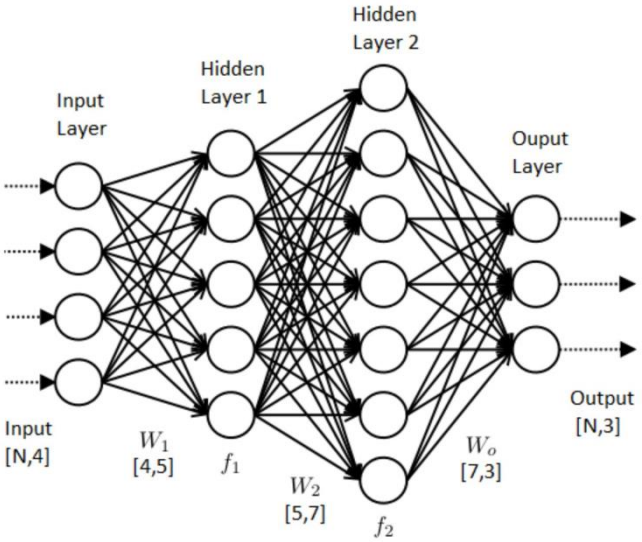

Figure 3. Artificial Neural Network (Source: GitHub, Author: Data Science lab of Berkley University)

The whole data set was divided into ten equal volumes which are called as ten folds. Nine folds datasets were used to train the model and one-fold dataset was used as test set using that model.

Tenfold cross validation was used to enhance the accuracy using all set of data equally to train the set. In this process, among the ten folds nine folds are used as training set and one set is used as test set. After predicting the class of test set the same is compared to its original class and the accuracy of the model is determined. The nine-fold datasets are shuffled randomly to use all the sets as both training and test set thereby averaging the result throughout the data.

V. METHODOLOGY

Input: User query.

Output: The answer to the query available in the repository and predicted by the system.

Step 1: Do preprocessing of the user query.

Step 2: POS tagging is done using 'Das & Halder' algorithm [Figure 4]

Step 3: Classify the query using the classification algorithms in to any one of the text categories.

Step 4: After determining the broad text category of the query apply the classification algorithms recursively to drill down to the portion of the text where probable answer exists.

Step 5: Use the knowledge base to form the answer.

Step 6: Return the answer to the user.

Algorithm 1: DAS & HALDER

Input: Bengali corpus to Shallow parser (LTRC)
Output: Inflected verbs are collected in a file Input.doc
Input to our system: Input.doc
Output of our system: Multiple files classified according to tense and person consisting of root form of verbs

```
1  begin
2    Corpus is taken as input to the Shallow parser
3    All the inflected verbs are collected in a file named as Input.doc
4    Input.doc is taken as input to our system
5    if বিভক্তি(in the Table 1) matches then
6      if person matches then
7        Verbs are passed to the respective method with the verbs and বিভক্তি for processing
8        Verbs are processed one by one
9        Panini's rules are applied on those verbs to extract root verbs
10       Root verb is stored in Output.doc file as well as separate file according to tense.
11     else
12       The verb collected from Shallow parser in Input.doc is not a verb
13   else
14     The verb collected from Shallow parser in Input.doc is not a verb
```

Figure 4. Das and Halder Algorithm

Tense	Type of tense	Suffices (বিভক্তি)
বর্তমান কাল (Present Tense)	সামান্য বর্তমান Simple Present	"ি","ে","েন","িস", "ই"
	ঘটমান বর্তমান Present Continuous	"ছি","িতেছি","ছে","িতেছে","ছ","িতেছ","ছেন"
	পুরাঘটিত বর্তমান Present Perfect	"েছি","িয়াছি","েছ","িয়াছ", "েছে","েছেন"
অতীত কাল (Past Tense)	সামান্য অতীত Simple Past	"লাম","লুম","িলাম","িলুম","লে","িলে","লেন","িলেন"
	ঘটমান অতীত Past Continuous	"ছিলাম","ছিলুম","িতেছিলাম","িতেছিলুম"
	পুরাঘটিত অতীত Past Perfect	"েছিলাম","েছিলুম" ,"িয়াছিলাম"
	নিত্যবৃত্ত অতীত Past Perfect Continuous	"তাম","তুম","িতাম","িতুম" ,"তে"
ভবিষ্যৎ কাল (Future Tense)	সামান্য ভবিষ্যৎ Simple Future	"ব","িব","বে","িবে", "বি","িবি"
	ঘটমান ভবিষ্যৎ Future Continuous	"তেথাকব","িতেথাকিব",তেথাকবে","িতেথাকিবে","তেথাকবি","িতেথাকিবি",
	পুরাঘটিত ভবিষ্যৎ Future Perfect	"েথাকব","িয়াথাকিব", "েথাকবে"

FIGURE 5. Table1: Different kind of suffices applied to the verb in Bengali with Tenses

Complete flowchart of the work is given in Figure 6. The POS tagging has been done by the shallow parser of IIIT, Hyderabad. As the shallow parser has been used directly by REST service so detailing of the POS tagging algorithm is not given. After POS tagging, identification of Function Words (FW), Content Words (CW), counting of FW and CW, identification of subjects, objects are done.

The detailing of the root verb extraction is given in [5]. Flowchart of root verb extraction and the required table is given in Figure 4 and 5 respectively.

Java is used to take the input query, process it and tokenize it. Mostly all the preprocessing work is also done using Java. Modules to extract root verb, to tag POS are called and result is collected. Vector creation for the classification is also done by Java from the normal text. Weka is used for classification.

WordNet is used to expand the dimensionality of the sentences in the repository by using similar words. The same technique is for expanding the dimensionality of the query using similar words as well.

Sense matching is improved by considering the WH words.

The knowledge Base is the relational data base implemented by PostgreSQL. After extracting the answer, rule based standard technique is used to form the answer in Bengali. The rules are mostly grammatical and help to form the answer sentence.

VI. RESULT

A. RESULT SUMMARY

A total number of 250 questions are fired. In 244 cases the perfect sentences are hit where the answers exist. It gives the 97.6 percent accuracy for hit.

In 214 number of cases correct answers are also generated with the help of knowledge base with acceptable accuracy percentage of 85.6.

In case of only root verb extraction 98 percent accuracy is achieved. The detailed result is presented in the next section.

B. DETAILED RESULT

i) Naïve Bayes

Classified Instances (Correct) - 88 percent
Classified Instances (Incorrect) -12 percent
Statistic (Kappa) - 0.197
Absolute error (Mean)- 0.467
Error (Root Mean Squared) - 0.501
Error (Relative Absolute) - 93.36 percent
Error (Root Relative Squared) -100.17 percent
Total Number of Instances-100

Naïve Bayes model identifies 88 correctly Classified Instances, 12 incorrectly classified instances. So, the accuracy is 88 percent and the inaccuracy is 12 percent.

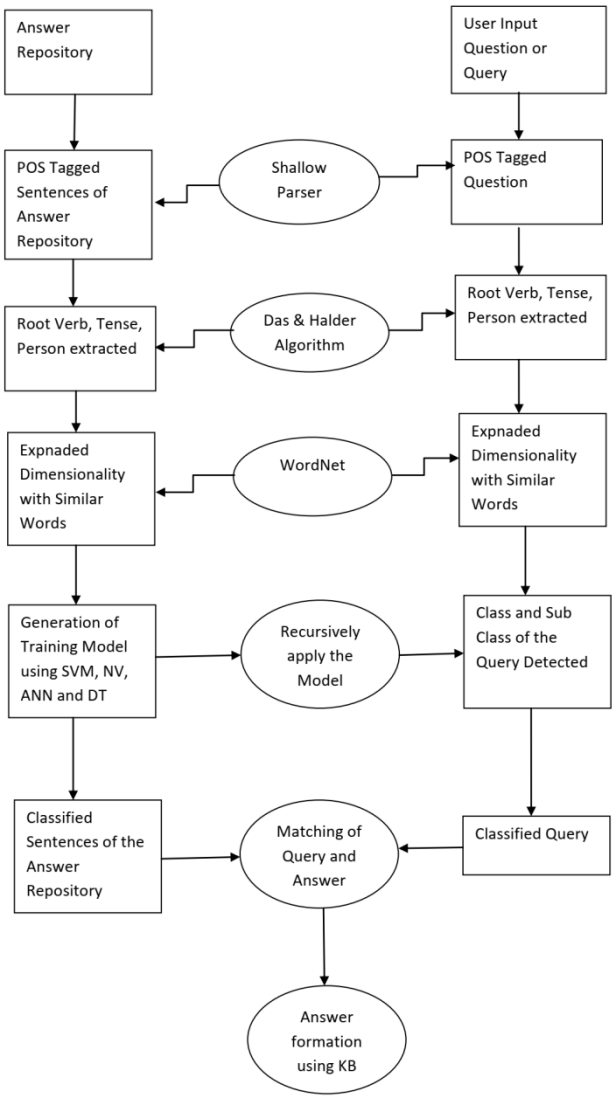

Figure 6. Detailed Flowchart of the Methodology Used

ii) SVM (SMO)

Classified Instances (Correct) - 86 percent
Classified Instances (Incorrect) - 14 percent
Statistic (Kappa) - 0.283
Absolute error (Mean) - 0.36
Error (Root Mean Squared) - 0.6
Error (Relative Absolute) - 72.062 percent
Error (Root Relative Squared) - 120.01 percent
Total Number of Instances-100

So, SVM model identifies 86 correctly classified instances and 14 incorrectly classified instances. The accuracy is 86 percent and the inaccuracy is 14 percent.

iii) ANN

Classified Instances (Correct) - 96 percent
Classified Instances (Incorrect) - 4 percent
Statistic (Kappa) - 0.1186
Absolute error (Mean) - 0.442
Error (Root Mean Squared) - 0.59
Error (Relative Absolute) - 8.478 percent
Error (Root Relative Squared) - 118.0054 percent
Total Number of Instances-100

So, the ANN model identifies 96 Correctly Classified Instances and 4 incorrectly classified instances. The accuracy is 96 percent and the inaccuracy is 4 percent.

iv) DECISION TREE

Classified Instances (Correct) - 73 percent
Classified Instances (Incorrect) - 27 percent
Statistic (Kappa) - 0.453
Absolute error (Mean) - 0.35
Error (Root Mean Squared) - 0.48
Error (Relative Absolute) - 69.92 percent
Error (Root Relative Squared) - 94.98 percent
Total Number of Instances-100

So, Decision Tree model identifies 73 Correctly Classified Instances and 27 incorrectly classified instances. The accuracy is 73 percent and inaccuracy is 27 percent.

VII. APPLICATIONS

One of the major applications of the semantic search is automatic question answering. In case of text chat or audio chat, now a days robots are able to answer the query. Siri in Apple products or Alexa of Amazon are the examples of such robots. Though both Siri and Alexa work primarily on syntactic search with web repository and personal database but intelligence is added with semantic search with newer versions.

News classification, summarization, storytelling, text generation are other applications where semantic search is used.

VIII. FUTURE SCOPE FOR IMPROVEMENTS

Semantic search using classification techniques are supervised learning techniques. Huge time and effort are invested to prepare the training set. Using unsupervised techniques and semi supervised techniques would save cost and time.

Knowledge base used in the system is static and grammatical rule based. Making the same dynamic or self-learning will improve the performance of the system.

Deep learning-based techniques will improve the result as well if it can incorporate the user feedback into the system.

IX. CONCLUSION

In the present work, we have tried to address the challenge of semantic search using classification techniques. Four classification techniques namely SVM, ANN, Naïve Bayes and Decision Tree are used for determining the class and subclass of the query. The result is fine grained using recursive classification. At the last stage, Knowledge Base is used to form the answer.

Primarily the work plan was targeted for Bengali language. Testing has been expanded for other Indian languages as well.

ACKNOWLEDGMENT

We sincerely express our gratitude to Professor N S Dash of the Language Research Unit (LRU) of ISI Kolkata for providing his valuable time to evaluate the performance of the system. He has also made available the corpus which is a product of TDIL project of Govt. of India. We would like to acknowledge the use of Shallow Parser which has been used for POS Tagging.

The Shallow Parser is developed by IIT Bombay and currently being hosted by IIIT Hyderabad.

REFERENCES

1. Deepak Gupta, Asif Ekbal, Surabhi Kumari, Pushpak Bhattacharyya: "MMQA: Multi Domain Multi-lingual Question-Answering Framework for English and Hindi", LREC 2018.
2. Iglesias. A. and Zhu. G., "Computing Semantic Similarity of Concepts in Knowledge graphs," Transactions on Knowledge and Data Engineering, vol. 29, no. 1, pp. 273 – 275, 2018.
3. Duhan. N., Nagpal. C. K., Katuria. M. and Payal G., "Semantic similarity between terms for query suggestion," in Proceedings of 5th ICRITO, vol. 2, no. 2, 2017, pp. 27 – 34.
4. Arijit Das and Diganta Saha, "Improvement of electronic governance and mobile governance in multilingual countries with digital etymology using sanskrit grammar," in Proc. IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Dhaka, Bangladesh, 2017, pp. 502 – 505.
5. Arijit Das, Tapas Halder and Diganta Saha, "Automatic extraction of Bengali root verbs using Paninian grammar" in Proc. 2nd IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT), Bangalore, India, 2017, pp. 953 – 956.
6. D. Gupta, P. Lenka, A. Ekbal, P. Bhattacharyya:" Uncovering Code-Mixed Challenges: A Framework for Linguistically Driven Question Generation and Neural Based Question Answering", CoNLL 2018: 119-130.
7. H. Dong. and F. K. Hussain, "Service-requester-centered service selection and ranking model for digital transportation ecosystems," Computing, vol. 97, no. 1, 79-102, 2015.
8. E. Portmann, "The FORA framework: a fuzzy grassroots ontology for online reputation management," Springer, USA, 2012.
9. H. Dong, "Semantic Search Engines and Related Technologies in: A Customized Semantic Service Retrieval Methodology for the Digital Ecosystems Environment," PhD Thesis, Curtin University, p. 71-104, 2010.
10. K. R. Chandu, M. K. Chinnakotla, A. W. Black, M. Shrivastava: WebShodh: A Code Mixed Factoid Question Answering System for Web. CLEF 2017: 104-111.
11. Adnan. G. S. M., T. Fatima,U. Habija, M. Illyass and Rashid, J., 2018. Matching Based Algorithm for Semantic Search and its Implementation. *Technical Journal*, *23*(04).
12. A. Allott., Y. Peng., C.H. Wei, K. Lee., L. Phan. and Z. Lu., 2018. LitVar: a semantic search engine for linking genomic variant data in PubMed and PMC. *Nucleic acids research*, *46*(W1), pp.W530-W536.
13. L. Sahni, A. Sehgal, S. Kochar, F. Ahmad and T. Ahmad, "A Novel Approach to Find Semantic Similarity Measure between Words," in Proc. IEEE 2nd International Symposium on Computational and Business Intelligence, 2014, pp 89-92.
14. J. Yunzhi., H. Zhou., H. Yang., Y. Shen., Z. Xie., Y. Yu. and F. Hang., "An Approach to Measuring Semantic Similarity and Relatedness between Concepts in An Ontology," in Proc. 23rd International conference on Automation and Computing(ICAC), 2017,pp 1-6.
15. B. Zhu., X. Li and J. B. Sancho, "A Novel Asymmetric Semantic Similarity Measurement for Semantic Job Matching," in Proc. International Conference on Security, Pattern Analysis and Cybernetics(SPAC), 2017,pp152-157.
16. H. Bast., B. Buchhold., and E. Haussmann., 2016. Semantic search on text and knowledge bases. *Foundations and Trends® in Information Retrieval*, *10*(2-3), pp.119-271.

AUTHORS PROFILE

ARIJIT DAS has received B.Tech. degree in CSE in 2011 from Govt. College of Engineering, Serampore, West Bengal and M.E. in CSE in 2013 from the Department of Computer Science and Engineering, Jadavpur University, Kolkata, India with GATE scholarship. Then he joined the public service as Scientific Officer in MeitY, Govt. of India. Currently he is doing his PhD (Engg.) in the Department of CSE, Jadavpur University, Kolkata. He is member of IEEE, ACM and life member of CSI. He has already published in reputed journals, books and conferences.

Dr. DIGANTA SAHA, PhD is a Professor of the Department of Computer Science and Engineering in Jadavpur University. He has research interest in NLP, Text Mining, Data Analytics.