ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

Statistical Translation using KNN and N-gram Algorithm for Machine Translation

Shilpa Modani , Priyanka More , Harshal Shedge , U. C. Patkar

Student, Dept. of Computer, Bharati Vidyapeeth's College of Engineering, Pune, India

Student, Dept. of Computer, Bharati Vidyapeeth's College of Engineering, Pune, India

Student, Dept. of Computer, Bharati Vidyapeeth's College of Engineering, Pune, India

H.O.D, Dept. of Computer., Bharati Vidyapeeth's College of Engineering, Pune, India

ABSTRACT:Statistical Machine Translation is used to translate one language to another language. It is one part of natural language processing. This system having three important models that is Language Model (LM), Translation Model (TM), Decoder. In this system we will generate an output as decoder and that uses language model and translation model. For this we will uses two languages such as English to Marathi. In this model translation is done by English language to Marathi language. For this we will use bigram, trigram, unigram and n-gram methods to find the probability. In this we generate the three results that are with KNN, without KNN, using both methods.

KEYWORDS: Machine translation, Corpus based machine translation, Machine learning, Information retrieval, N-gram ,Predictive Analytic, Multilingual chatter, Statistical machine translation ,KNN

I. INTRODUCTION

Now a day's automatic natural language translator such as google translator is widely used by peoples around the world. Many time's people observe that some incorrect, disappointing result by these translation methods. The translation result varies depending on the language used. In this paper, we calculate the accuracy with knn and n-gram algorithm on the basis of theoretical n practical knowledge. We compare the result of these two algorithms with each other to know the better one. and also, we take them combine to calculate accuracy.

KNN algorithm

The KNN algorithm is used for classification and regression in machine learning. KNN store whole dataset so there is no need of learning. Efficient implementations can store the data using complex data structure to matching of new patterns during prediction efficient. In KNN the prediction is done directly by the dataset. Predictions are made for new instances by searching through entire training set for K most similar neighbour's and summarizing the output text for those K neighbour's. The distance measure is used to determine which of the K-Neighbours in the training set are most similar to a new input. And the Euclidean distance is used for real valued input variables.
Calculation by Euclidean distance:
Euclidean_Distance (x, xi) = sqrt (sum ((xj – xij)^2))
Here,
(x) -> New point
(xi) ->Existing point
(j) ->All input attributes
There are many distances are used like Euclidean distance, i.e. Hamming distance, Manhattan Distance, Minkowski Distance. You can choose the best distance matrix based on the properties of your data. If you are confused, then you can experiment with different distance metrics and different values of K together and see which matric gives most

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

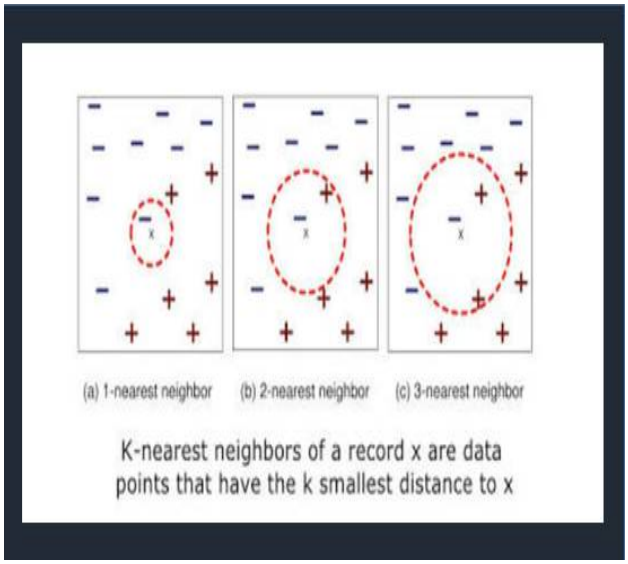
(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

accurate models. Euclidean is best if input variables are similar in type. When KNN is used in translation, the output can be calculated as the class with the highest frequency from K-most similar instances.

KNN is the lazy learning algorithm which has no parameter. As KNN is non-parametric it means it does not make any assumption on the data selection or matching. This algorithm is lazy that means it uses the training data set at the time of testing phase. This makes decision based on the entire training data set. KNN assumes that the data is in future space, since they have a distance notation. Each of the training data consist of a set of vectors associated with each vector, it will be either + or -. The K number decides how many neighbors influence in classification. If the no. of classes is 2. And k=1 then the algorithm simply called as nearest neighbor algorithm.



Fig[1]

fig.[1] shows the KNN algorithms working. As we know the KNN finds the nearest neighbors in the training dataset to find the efficient match. In the fig[1] the first stage is to find all the nearest neighbor like 1, 2,3 ..n and then calculate the distance between all the neighbors from the from data point by distance matric. Simply the nearest one will be assumed as the perfect match for the dataset.

N-GRAM Algorithm

The N-gram algorithm are predominant in the natural language processing (NLP) and its application. N-gram are sequence of items as they appear in the text/document, These items can be phonemes, letters, words, etc. when the items are word, then the n-gram also be called singles or uni-gram. And when the items are of two word then it can also be called as bi-gram and so on. So, the continues sequence of n words from a given sequence of text or speech are called as n-gram. The n-gram is now widely used in probabilistic, communication theory, Statistical natural language processing, etc. It is a type of probabilistic language model used for predicting the next item in sequence of items.
The n-gram also be used in sequence of efficient matching text/documents. This is used by converting the sequence of elements to a set of n-gram, thus allowing the sequence to be compared to other sequences in an efficient manner. For example, if we are converting the string into English alphabets, suppose both the strings are ‘pqr’ and ‘qrp’ gives rise to exactly same bi-gram ‘qr’ is clearly not same as ‘qr’ and ‘rp’. However, we know that if two strings have similar vector representation then their meaning are little similar. So, this algorithm is also used in machine learning for language translation. But this method is not same as the KNN.

The n-gram algorithm matches the sequence of words with the training data set. The fig[2] shows the example of working of n-gram algorithm. In which one sentence is given as input and we have to match that sentence with training set n give the efficient translation for that set of words.

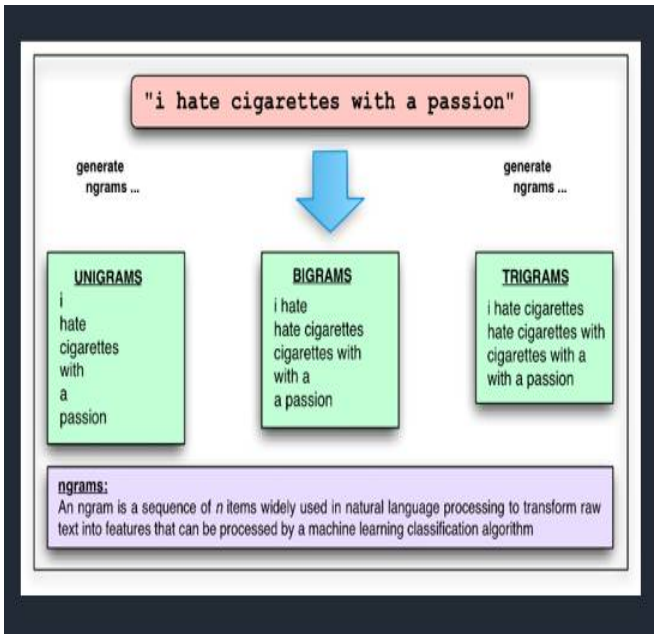
ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017



Fig[2]

These sentence generate in unigram, bigram, trigram, n-gram to get the efficient match. Because there are three types of words over the different categories. Stop words lie are, a, an, the,.. these words appear in almost all documents. Common words that may appear in any news category but they are not English words happen, problems,... The words that differ categories like football in sports, director in media,… and these are the words that help the classifier most to learn. Using unigram the single words are calcified that is in category 1 and the bigram and trigram are used where the more than one word is need to use for classification like happened in football appears in only sports. So this algorithm gives the more efficient result than KNN.

II. PROPOSED ALGORITHM

A. *Design Considerations:*

- Insert the data into the database.
- Create grammar rules as per the language used.
- Make training and testing of the inserted data.

B. *Description of the Proposed Algorithm:*

Aim of the proposed algorithm is to maximize the conversion rate of text from one language to another. The proposed algorithm is consists of two main steps.

Step 1: Training phase:

The training phase consist of training the inserted data into the database for the further use. The data is compared with both the languages and train it for conversion.

Step 2: Testing phase:

The testing phase is to test the data which train by the training phase. It chakes the conversion of data is correct or not and the percentile of converted data as follows,

$m' = \max\_ m\ P(e/m)$ …. (1)

Where,

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

P(e/m) = Probability in file translation.
e = Source language
m = Possible translation of targeted language

By the Bayes theorem we can represent P(m/e) as shown in eq. 2

$$P(e/m) = \frac{P(e)P(m/e)}{P(m)} \quad \ldots. (2)$$

By using eq. (1) and (2) we get eq. (3) as follow,

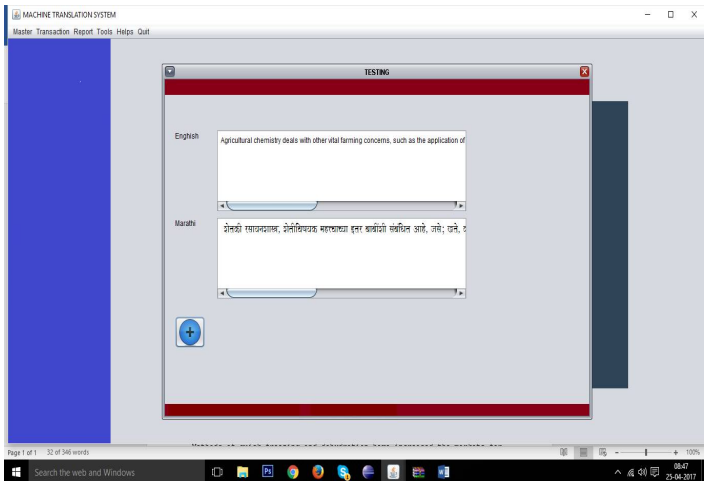
$$m' = \max_m \frac{P(e)P(m/e)}{P(e)}$$

III. PSEUDO CODE

Step 1: Create database.
Step 2: Insert data for both the languages.
Step 3: Train the inserted data as per the grammar rules.
Step 4: Test the train data using eq. above.
Step 5: Calculate the percentile of correct conversion of text.
Step 6: go to step 3.
Step 7: End.

IV. SIMULATION RESULTS

The simulation studies involve the deterministic small data inserted into the database. The result is shown as per our proposed work. The proposed work is as we seen the KNN selects the word nearest one and the n-gram matches the sequence of items with training data set for efficient result. Then we are combining both the algorithms for better and most accurate result. The n-gram will make categories of data and the KNN will use nearest one. We proposed to combine these two algorithms for language translation. The result of our work is shown as below,



ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

As you see in the above output window there is one English corpus is entered in English box and exactly below the translation of above content is displayed.

V. CONCLUSION AND FUTURE WORK

The simulation results showed that the proposed algorithm performs better with the total database inserted into the system. The proposed algorithm provides better way to convert the text from English language to Marathi language . The result shows that the result is better if the database is strong.

REFERENCES

1. B.N.V Narasimha Raju, M S V S Bhadri Raju," ***Statistical Machine Translation System for Indian Languages"*** 2016 IEEE 6th International Conference on Advanced Computing
2. T Siddiqui, U. S. Tiwary, "Natural Language Processing and Information retrieval," Oxford Press, 2008.
3. Sanjay Kumar Dwivedi and P PSukhadeve, "Machine Translation System in Indian Perspectives," J. Computer Sci., Vol. 6, Issue No. 10, pp. 1111-1116, 2010.
4. D.D. Rao, "Machine Translation A Gentle Introduction",
RESONANCE, July 1998.
5. Adam Lopez, "Statistical Machine Translation," ACM Computing Surveys, Vol. 40, Issue No. 3, Article 8, August 2008.
6. B N V Narasimha Raju et.al, "Translation Approaches in Cross Language Information Retrieval," In Proceedings of International Conference on Computer and Communications Technologies (ICCCT), publisher is IEEE, 2014.