Text Segmentation for Online Subjective Examination Using Machine Learning

Shahid Khan[1], Rakshanda Chavan[2] , Diksha Singh[3], Tina Sajwan[4]

[1,2,3,4] *Modern Education Society's College of Engineering, Pune-411001*

---***---

Abstract - *This paper focuses on text segmentation for natural language using k-Nearest Neighbour (K-NN) classifier , which is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The Text segmentation divides written text into meaningful units, which is used by humans when reading text, and artificial processes implemented in computers which are subject to natural language processing. K-NN computes the similarity measure among attributes to determine similarity between feature vectors after which K-NN is modified based on the similarity measure, this version is applied into the text segmentation task. The goal of this paper is to implement natural language processing using text segmentation which provides the benefits.*

Key Words: K-NN, text segmentation, feature similarity, NLP

1.INTRODUCTION

The text segmentation is defined as process of segmenting automatically a large text into many parts based on its topic or content. The information retrieval (IR) systems tend to retrieve long texts which contain more than one topic, as very high relevant texts to the given query, so the long texts need to be segmented into text partitions topic by topic. The task of text segmentation is to partition the text into sentences and paragraphs and judge whether the topic boundary is put or not between two adjacent sentences or paragraph. In this task, the text is given as the input and segmented into paragraphs, a list of pairs of adjacent paragraphs is generated, and each pair is judged whether we put the topic boundary between them, or not. The task is interpreted into a binary classification where each pair of paragraphs is classified into separation or non-separation. The task may be interpreted into the binary classification where each sentence or paragraph pair into the transition to the different topic or the continuation of the identical topic.

Some issues are caused by encoding texts into numerical vectors and computing their similarities based on only attribute values. This problem causes very high costs for processing each numerical vector representing a document in terms of time and system resources. Much more training examples are required proportionally to the dimension for avoiding overfitting. The second problem is sparse distribution where each numerical vector has zero values dominantly.

Let us mention what we propose in this research as some agenda. In this research, we assume that words are given as features of numerical vectors in encoding texts, and they have their semantic relations with others. Based on the assumption, we define the similarity measure for computing the similarity between feature vectors, considering both feature values and features. We modify the KNN into the version where both the feature similarity and the feature value similarity are used, and apply it to the classification task mapped from the text segmentation. As benefits from this research, we expect its more tolerance to the sparse distributions and the potential avoidance of the huge dimensionality.

Let us mention what is expected from this research as benefits by implementing the above ideas. We may cut down the dimensionality in encoding texts into numerical vectors, potentially. The information loss in computing the similarity between texts may be reduced by reflecting the similarities among the features.

We present some benefits which are expected from this research. By representing the texts into alternative one to the numerical vectors, we may escape from the two main problems in doing so. The proposed approach becomes less sensitive to the sparse distribution of numerical vectors, because the similarity among features is captured as well as among feature values.

2. RELATED WORK

Let us survey the previous cases of encoding texts into structured forms for using the machine learning algorithms for text mining tasks. The three main problems, huge dimensionality, sparse distribution, and poor transparency, have existed inherently in encoding them into numerical vectors. In previous works, various schemes of pre-processing texts have been proposed, in order to solve the problems.

In paper [1], it is given that text segmentation refers to the process of segmenting an article into its several parts based on its content. Because in the information retrieval systems, a long text tends to be retrieved most frequently by overestimation of its relevancy to a query, we need to segment it into its several parts, in order to avoid the problem. In this task, the text is given as the input and segmented into paragraphs, a list of pairs of adjacent paragraphs is generated, and each pair is judged whether we put the topic boundary between them, or not.

In paper [2], The task of text segmentation is to partition the text into sentences and paragraphs and judge whether the topic boundary is put or not between two adjacent sentences or paragraph. The task may be interpreted into the binary classification where each sentence or paragraph pair into the transition to the different topic or the continuation of the identical topic. Segmentation of speech texts into sentences or paragraphs may be considered but covered in the next research. In the text categorization, the sample texts may span over various domains, whereas in the text segmentation, the sample paragraphs should be within a domain. Therefore, although the text segmentation belongs to the classification task, it should be distinguished from the topic based text categorization. The text segmentation is mapped into a binary classification.

In paper [3], the application of the back propagation to the judgment of keywords is validated restrictedly. The definition of the back propagation to the judgment of keywords may be considered in various ways. The Information systems dealing with documents, such as Knowledge Management (KM), Information Retrieval (IR) and Digital Library (DL) systems require the storage of documents and structured data, called the document surrogate, associated with documents. Documents are written in natural language and cannot be processed directly by computers. A typical document surrogate, which is converted from the natural language document by computer, contains indices of the document and includes main words reflecting the contents. Indexing defines the process of converting a document into a list of words included in it. This paper proposed the application of back propagation and consideration of more factors with the addition to TF (Term Frequency) and IDF (Inverse Document Frequency).

Paper [4], states that text categorization is the process of assigning one or some among predefined categories to each document. The task belongs to pattern classification where texts or documents are given as patterns. Note that almost information in any system is given as textual formats dominantly over numerical one. For managing efficiently the kind of information given as the textual format, techniques of text categorization are necessary; text categorization became a very interesting research topic in both academic and industrial worlds. In this version of the proposed text categorization system, the number of entries of tables is fixed constantly. The proposed one is called static index based approach. However, the optimal number of entries is very dependent on the given document or corpus. The size of each table should be optimized in terms of two factors: reliability and efficiency.

In paper [5], authors tried to understand the automated categorization (or classification) of texts into predefined categories has witnessed a booming interest in the last 10 years, due to the increased availability of documents in digital form and the ensuing need to organize them. It is important to bear in mind that the considerations above are not absolute statements (if there may be any) on the comparative effectiveness of these TC methods. One of the reasons is that a particular applicative context may exhibit very different characteristics from the ones to be found in Reuters, and different classifiers may respond differently to these characteristics. An experimental study by Joachims [1998] involving support vector machines, k-NN, decision trees, Rocchio, and Naive Bayes, showed all these classifiers to have similar effectiveness on categories with 300 positive training examples each. The fact that this experiment involved the methods which have scored best (support vector machines, k-NN) and worst (Rocchio and Naive Bayes).Most popular approach to TC, at least in the operational (i.e., real world applications) community, was a knowledge engineering (KE).

In paper [6], the authors have studied that text clustering refers to the process of segmenting a particular group of documents into sub groups each of which contains content based similar documents. A collection or group of documents is given as the input of the task. Several smaller groups of content-based similar documents are generated from the task as its output. Although there are many heuristic approaches to the task, unsupervised learning algorithms have been used as state of the art approaches to it. The process of encoding documents into numerical vectors for using traditional unsupervised learning algorithms for text clustering causes the two main problems. The first problem is huge dimensionality where documents must be encoded into very large dimensional numerical vectors for preventing information loss. In general, documents must be encoded at least into several hundreds dimensional numerical vectors in previous literatures. This problem causes very expensive cost for processing each numerical vector representing a document in terms of time and system resources. Furthermore, much more training examples are required proportionally to the dimension for avoiding overfitting.The second problem is sparse distribution where each numerical vector has zero values dominantly. In other words, more than 90 degree 0 of its elements are zero values in each numerical vector. This phenomenon degrades the discrimination among numerical vectors. This causes poor performance of text categorization or text clustering. In order to improve performance of both tasks, the two problems should be solved.

3. PROPOSED SYSTEM

KNN Classifier:

This section tells about the KNN classifier which is an algorithm used for text segmentation. It keeps the record of all the previous cases and another unknown case is been classified. It is a type of supervised learning. The unknown case is been classified by the maximum votes of its K nearest neighbours. It is a kind of Machine Learning algorithm and also one of the simplest algorithm used for classification. It considers the similarity between the attributes of the answers written by the user and then computes the similarities between the features of the answer and specimen answers. In this research, we encode sentence pairs or paragraph pairs into string vectors, and apply the string vector based version of KNN to the classification task mapped from the text segmentation

NLP:

This section is concerned with Natural Language Processing which is a field of AI(Artificial Intelligence). It is about the co-operation between the computer and the Natural Language used by humans. NLP is helpful in solving many problems like machine translation and text segmentation.

Text Segmentation:

This section is concerned about Text segmentation which is the process where the text which is been written is divided into small parts. The term applies both to mental processes used by humans when reading text, and to artificial processes implemented in computers, which are the subject of natural language processing. It is very helpful in assisting computers so that it is possible for the computers to do artificial things. It is a precursor Natural Language Processing. Text Segmentation recognizes the boundaries in between the words.

Data Store:

This section tells us about the role of data store in the process. A data store is a repository for storing collections of data, such as database. A data store is basically a connection to the repository of data, whether the data is stored in a single database or in one more different files. The data store can be used to gain data or you can export the data from results and then store it in the data store, or both. The data collected from the users is stored in the data store. For the processing the data stored in the data store is processed and stored back into the data store for the users to retrieve their processed data whenever he wants. Hence data store plays a major role in the entire process. For the data to be stored in the data store it need not compulsorily be arranged in some relational format.

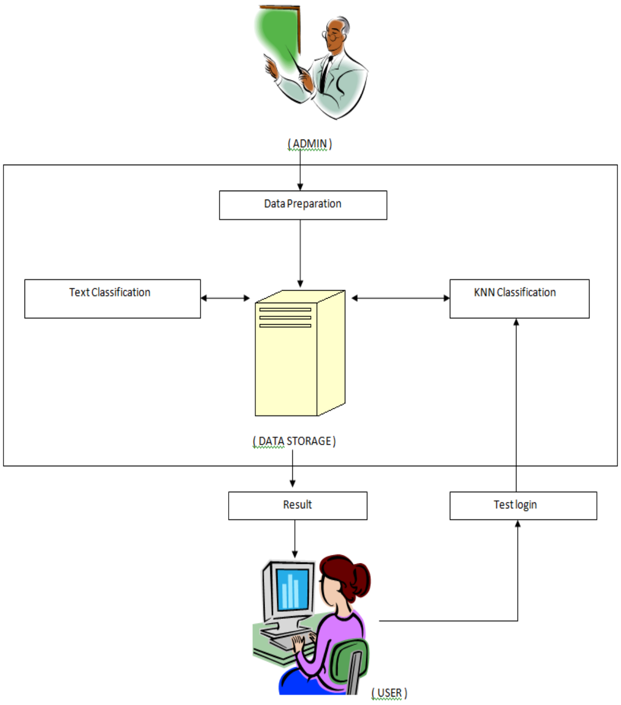

Fig -1: Architecture Diagram

4. CONCLUSIONS

An examination system is developed based on the web. This paper describes the principle of the system, presents the main functions of the system, analyzes the auto-generating test paper algorithm, and discusses the security of the system. With the help of the algorithm we can conduct online subjective exams anywhere and everywhere.

It saves time as it allows number of students to give the exam at a time and displays the results as the test gets over, so no need to wait for the result. It is automatically generated by the server. Staff has a privilege to create, modify and delete the test papers and its particular questions. Student can register, login and give the test with his specific id, and can see the results as well.

ACKNOWLEDGEMENT

We thank our guide Prof. A. D. Dhawale for his guidance and support.

REFERENCES

1) Taeho Jo, “Using K Nearest Neighbors for Text Segmentation with Feature Similarity”, International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE), 2017.

2) Taeho Jo, “Content based Segmentation of Texts using Table based KNN”, IKE, 2017.

3) Taeho Jo, Malrey Lee , and Thomas M Gatton, “Keyword Extraction from Documents Using a Neural Network Model”, IEEE, 2016.

4) T. Jo, “NTC (Neural Text Categorizer): Neural Network for Text Categorization”, pp83-96, International Journal of Information Studies, Vol 2, No 2, 2010.

5) T. Jo, “Normalized Table Matching Algorithm as Approach to Text Categorization”, pp839-849, Soft Computing, Vol 19, No 4, 2015.

6) T. Jo, “Single Pass Algorithm for Text Clustering by Encoding Documents into Tables”, pp1749-1757, Journal of Korea Multimedia Society, Vol 11, No 12, 2008.

7) T. Jo and D. Cho, “Index based Approach for Text Categorization”, International Journal of Mathematics and Computers in Simulation, Vol 2, No 1, 2008.

8) H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, “Text Classification with String Kernels”, pp419-444, Journal of Machine Learning Research, Vol 2, No 2, 2002.

9) F. Sebastiani, “Machine Learning in Automated Text Categorization”, pp1-47, ACM Computing Survey, Vol 34, No 1, 2002

10) T. Jo, “Representation of Texts into String Vectors for Text Categorization”, pp110-127, Journal of Computing Science and Engineering, Vol 4, No 2, 2010.