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 4, April 2017

Review on Query Focused Summarization using TF-IDF, K-Mean Clustering and HMM

Sonali Gandhi, Praveen Sharma

M. Tech (pursuing), Dept. of CSE, NGF College of Engineering , Palwal, Haryana under the Affiliation of Maharshi Dayanand University at Rohtak, Haryana – India

Assistant Professor, Dept. of CSE, NGF College of Engineering, Haryana under the Affiliation of Maharshi Dayanand University at Rohtak, Haryana - India

ABSTRACT: Numerous approaches for distinguishing necessary content for automatic text summarization are developed till date. Query focused summarization illustration approach is first derive from an intermediate illustration of the text that captures the topics mentioned within the input and supported these illustration of topics, sentences within the input document whereas impact or influence measure scored for importance factors are be calculated. In distinction with machine learning indicator illustration approaches, the text is represented by a various set of doable indicators of importance that inculcate at discovering interestingness. These indicators of machine learning approached measure and combined various techniques which finally optimize the text based on various choices and select the most effective set of sentences to create a outline ort summary. Subsequently, in this scheme we propose the effective technique using TF-IDF, K-Mean Clustering and Hidden Markov Model with amalgamation to produce enhances better Query Focused Summarization Model for better ready reference and perusal.

In query-focused summarization, the importance of each sentence will be determined by a combination of two factors: how relevant is that sentence to the user question and how important is the sentence in the context of the input in which it appears. There are two classes of approaches to this problem. The first adapts techniques for generic summarization of news. For example, an approach using topic signature words is extended for query-focused summarization by assuming that the words that should appear in a summary have the following probability: a word has probability zero of appearing in a summary for a user defined topic if it neither appears in the user query nor is a topic signature word for the input; the probability of the word to appear in the summary is five percent if it either appears in the user query or is a topic signature, but not both; and the probability of a word to appear in a summary is, if it is both in the user query and in the list of topic signature words for the input. These probabilities are arbitrarily chosen, but in fact work well when used to assign weights to sentences equal to the average probability of words in the sentence. Cluster based approaches have also been adapted for query-focused summarization with technical modifications. In the scheme we propose new mechanism with existing artifacts for identifying relevant and salient sentences.

KEYWORDS: Term Frequency – Inverse Document Frequency (TF-IDF), Machine Learning (ML), Web Mining, K-Mean Clustering, Hidden Markov Model (HMM).

I. INTRODUCTION

The information retrieval techniques are the most popular techniques used for the most relevant information retrieval. It is very much crucial to get most appropriate queries when the user enters the query. To achieve the required tasks, the approach pre mines the internet to retrieve the potential cluster of queries followed by finding the most popular queries in cluster. The output of both mining processed is utilized to return relevant pages to the users while recommending him with popular focused queries, consequently we are going to discuss the proposed nitty-gritty as under:

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 4, April 2017

TF-IDF weighting: Term Frequency-Inverse Document Frequency The word probability approach relies on a stop word list to eliminate too common words from consideration. Deciding which words to include in a stop list, however, is not a trivial task and assigning TF*IDF weights to words provide a better alternative. This weighting exploits counts from a background corpus, which is a large collection of documents, normally from the same genre as the document that is to be summarized; the background corpus serves as indication of how often a word may be expected to appear in an arbitrary text. The only additional information besides the term frequency c(w) that we need in order to compute the weight of a word w which appears c(w) times in the input for summarization is the number of documents, d(w), in a background corpus of D documents that contain the word. This allows us to compute the inverse document frequency the figure 1 depicts the model.

$$TF * IDF_w = c(w).\log \frac{D}{d(w)}$$

Figure 1: TF*IDF formulation

In many cases c(w) is divided by the maximum number of occurrences of any word in the document, which normalizes for document length. Descriptive topic words are those that appear often in a document, but are not very common in other documents. Words that appear in most documents will have an IDF close to zero. The TF-IDF weights of words are good indicators of importance, and they are easy and fast to compute. These properties explain why TF-IDF is incorporated in one form or another in most current systems. Sometimes another method is used which combines the term frequency with the inverse document frequency (TF-IDF). The TF-IDF weight is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. Variations of the TF-IDF weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query.
Typically, the TF-IDF weight is composed by two terms: the first computes the normalized Term Frequency (TF), aka. the number of times a word appears in a document, divided by the total number of words in that document; the second term is the Inverse Document Frequency (IDF), computed as the logarithm of the number of the documents in the corpus divided by the number of documents where the specific term appears.

TF: Term Frequency, which measures how frequently a term occurs in a document. Since every document is different in length, it is possible that a term would appear much more times in long documents than shorter ones. Thus, the term frequency is often divided by the document length as a way of normalization:

TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document).

IDF: Inverse Document Frequency, which measures how important a term is. While computing TF, all terms are considered equally important. However it is known that certain terms, such as "is", "of", and "that", may appear a lot of times but have little importance. Thus we need to weigh down the frequent terms while scale up the rare ones, by computing the IDF for t, the number of terms is given below:

IDF(t) = log_e(Total number of documents / Number of documents with term t in it).

The document frequency df_i is the number of documents in a collection of N documents in which the term t_i occurs. A typical inverse document frequency (idf) factor of this type is given by $log\left(\frac{N}{df_i}\right)$. The weight of a term t_i in a document is given by:

$$w_i = tf_i \times log\left(\frac{N}{df_i}\right)$$

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 4, April 2017

Hidden Markov Mode: Markov Models are a powerful abstraction for time series data, but fail to capture a very common scenario. How can we reason about a series of states if we cannot observe the states themselves, but rather only some probabilistic function of those states? This is the scenario for part-of-speech tagging where the words are observed but the parts-of-speech tags aren't and for speech recognition where the sound sequence is observed but not the words that generated it. In an HMM, we assume that our data was generated by the following process: posit the existence of a series of states ~z over the length of our time series. This state sequence is generated by a Markov model parameterized by a state transition matrix A. At each time step t, we select an output x_t as a function of the state z_t. Therefore, to get the probability of a sequence of observations, we need to add up the likelihood of the data ~x given every possible series of states below figure 2 depicts the same.

$$P(\vec{x}; A, B) = \sum_{\vec{z}} P(\vec{x}, \vec{z}; A, B)$$
$$= \sum_{\vec{z}} P(\vec{x}|\vec{z}; A, B) P(\vec{z}; A, B)$$

Figure 2: HMM formulation

K-Mean Clustering: K-mean clustering is the part of Partitioning Clustering analysis which aims to form k groups from the n data points taken as an input. This partitioning happens due to the data point associating itself with the nearest mean.

Classical Approach - The main steps of k-means algorithm are as follows:

1. Randomly select k data points to represent the seed centroids.
2. Repeat steps 3 and 4 until cluster membership stabilizes- either number of iterations specified by the user, or the dimensions of centroid does not change.
3. Generate a new partition by assigning each data point to its closest cluster center - assigning happens based on the closest mean.
4. Compute new cluster centres - calculating new centroids using the mean for multidimensional data-points

Direct K-Means Clustering:

K-means document clustering comes under partition technique of clustering where one-level (un-nested) partitioning of the data points is created. If K is the desired number of clusters, then partition approaches typically find all K clusters at once. K-means is based on the idea that a center point can represent a cluster. In particular, for K-means we use the notion of a centroid, which is the mean or median point of a group of points. Basic k-means algorithm is given below :-

Input: K: number of cluster, D: Top N documents
Output: K clusters of documents
Algorithm:
Step.1 Generate K centroids C1, C2, ..,Ck by randomly choosing K documents from D Repeat until there is no change in cluster between two consecutive iterations.
Step.2 for each document di in D
for j = 1 to K Sim(Cj, di) = Find cosine similarity between di and Cj
end for
Assign di to cluster j for which Sim(Cj, di) is maximum
end for
Step.3 Update centroid for each cluster
end loop
Step.4 end K-Means

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 4, April 2017

II. LITERATURE REVIEW

R. Baeza-Yates, C. Hurtado, and M. Mendoza [6] suggests that, the search engine gives the list of related results. These results are based on the previously searched queries or such technique can be used to tune or redirect the user. In this method the clustering algorithm is used. The clustering is done on the basis of previously fired queries. It clusters the semantically similar queries. It does not only give the clustered data but it also ranks the suggested list of result. The ranking is done on the basis of two conditions, 1. Similarity of queries to the input query 2. Observation that measures the attention of the user attracted towards the result of the query. The combination of both these conditions measures the user interests. In the given algorithm, query session is considered for giving the result. The query session also considers the rank of clicked URL. The relevance ranking is measured by using two components similarity of query and support of query.

Harshada P. Bhambure, MandarMokashi[9] discuses that user search goals for a query by clustering feedback sessions. For that, we use a concept of pseudo document, which is the revised version of feedback session. At the end, we cluster these pseudo-documents to infer user search goals and represent them with some keywords. Since the evaluation of clustering is also an important problem, we used evaluation criterion classified average precision (CAP) to evaluate the performance of the restructured web search results. The clustering is done by bisecting k means where in the existing system it is done by k means clustering. The new algorithm increases the efficiency of result. After the segmented result formation, the result in the every segment is reorganized as per number of clicks of URLs. The link which is clicked more number of times will appear at first location in the segment. This reduces the time requirement for searching.

DasariAmarendra, KavetiKiranKumar[10] suggest that user's information needs due to the use of short queries with uncertain terms. thus to get the best results it is necessary to capture different user search goals. These user goals are nothing but information on different aspects of a query that different users want to obtain. The judgment and analysis of user search goals can be improved by the relevant result obtained from search engine and user's feedback. Here, feedback sessions are used to discover different user search goals based on series of both clicked and unclicked URL's. The pseudo-documents are generated to better represent feedback sessions which can reflect the information need of user. With this the original search results are restructured and to evaluate the performance of restructured search results, classified average precision (CAP) is used. This evaluation is used as feedback to select the optimal user search goals.

BhaveshPandya, CharmiChaniyara, DarshanSanghavi, KrutarthMajithia[11] suggest that ambiguous query, different users may have different search goals when they submit it to a search engine. The inference and analysis of user search goals can be very useful in improving search engine relevance and user experience. In this propose a novel approach to infer user search goals by analysing search engine query logs a framework to discover different user search goals for a query by clustering the proposed feedback sessions. Feedback sessions are constructed from user click through logs and can efficiently reflect the information needs of users a novel approach to generate pseudo-documents to better represent the feedback sessions for clustering. Classified Average Precision (CAP) to evaluate the performance of inferring user search goals. Experimental results are presented using user click through logs from a commercial search engine to validate the effectiveness.

III. PROPOSED WORK

In the proposed scheme we will classify sentences to a K-Mean Clustering which captures the theme of the sentence and then calculate a similarity measure between the sentence and the document that it belongs to. Our approach uses IF and IDF along with Hidden Markov Model. The proposed approach involves the amalgamation of all three various model into one nitty-gritty figure 3 depicts the scheme.

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 4, April 2017

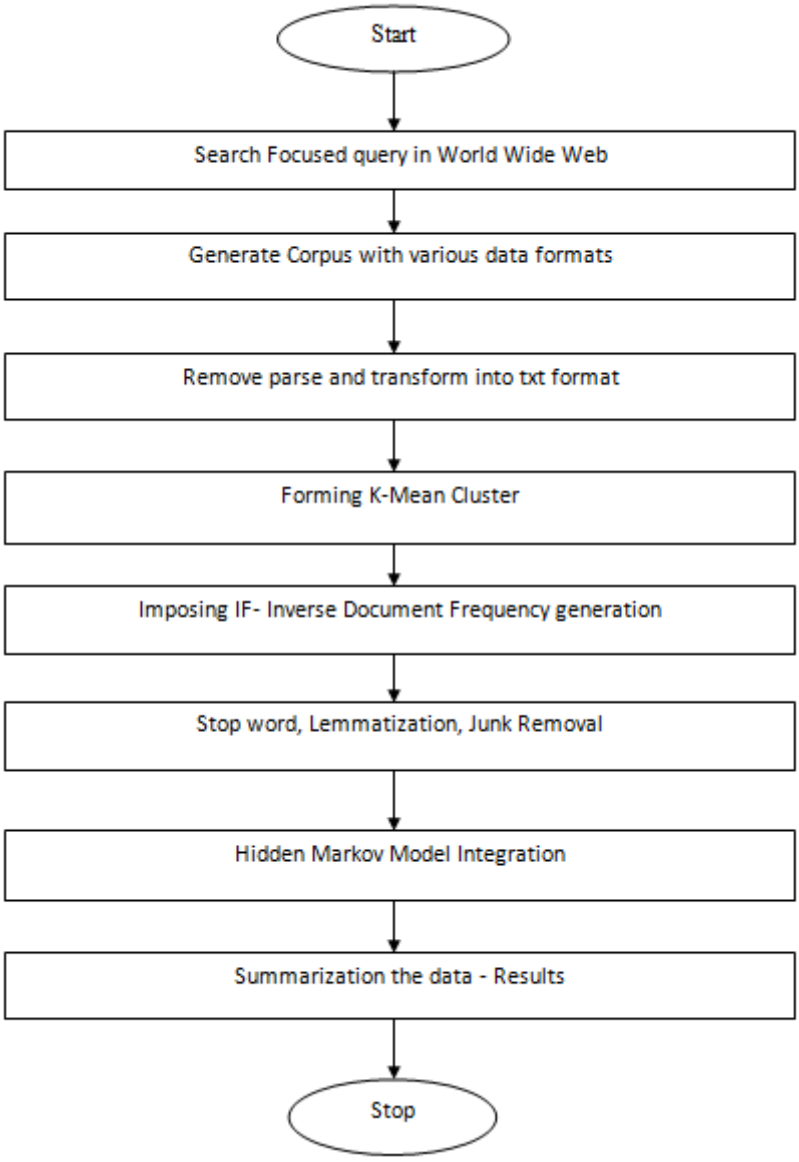

Figure 3: Flow Diagram of Proposed Scheme

REFERENCES

1. I. Mele, " Web Usage Mining for Enhancing Search –Result Delivery and Helping Users to Find Interesting Web Content," ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '13), pp. 765-769, 2013
2. P. Sudhakar, G. Poonkuzhali, R. Kishor Kumar, "Content Based Ranking for Search Engines," Proc. International Multi Conference of Engineers and Computer Scientists (IMECS 12), 2012.
3. H. Chen and S. Dumais, "Bringing Order to the Web: Automatically Categorizing Search Results," Proc. SIGCHI Conf. Human Factors in Computing Systems (SIGCHI '00), pp. 145-152, 2000.
4. X. Wang and C.-X Zhai, "Learn from Web Search Logs to Organize Search Results," Proc. 30th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '07), pp. 87-94, 2007.
5. H.-J Zeng, Q.-C He, Z. Chen, W.-Y Ma, and J. Ma, "Learning to Cluster Web Search Results," Proc. 27th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '04), pp. 210-217, 2004.

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 4, April 2017

6. R. Baeza-Yates, C. Hurtado, and M. Mendoza, "Query Recommendation Using Query Logs in Search Engines," Proc. Int'l Conf. Current Trends in Database Technology (EDBT '04), pp. 588-596, and 2004.
7. H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li, "Context-Aware Query Suggestion by Mining Click Through," Proc. 14th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (SIGKDD '08), pp. 875883, 2008.
8. Harshada P. Bhambure, Mandar Mokashi, "Inferring User Search Goals Using Feedback Session" Conf. Research paper, pp.2319-7064 and 2013.
9. DasariAmarendra, KavetiKiran Kumar, "Inferring User Search Goals with Feedback Sessions using K-means clustering algorithm", Volume 2, Issue 11,pp. 780-784,November-2015.
10. Bhavesh Pandya et al., "A New Algorithm for Inferring User Search Goals with Feedback Sessions", Int. Journal of Engineering Research and Applications www.ijera.com ISSN: 2248-9622, Vol. 5, Issue 8, (Part - 2), pp.30-33, August 2015.
11. Fang Chen, Kesong Han and Guilin Chen, "An Approach to Sentence Selection Based Text Summarization", In the Proceedings of IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering, Volume: l,pp.489-493,2002.
12. Ben Hachey, "Multi-Document Summarization Using Generic Relation Extraction", Proceedings of the Conference on Empirical Methods in Natural Language Processing, pages 420-429, 2009.
13. Alok Ranjan Pal, Projjwal Kumar Maiti and Dig anta Saha, "An Approach To Automatic Text Summarization Using Simplified Lesk Algorithm And WordNet", International Computer Modelling (IJCTCM)Journal of Control Theory and Vol.3, No.4/5, September 2013
14. A.R.Kulkarni, S.S.Apte, an automatic text summarization using lexical cohesion and correlation of sentences, International Journal of Research in Engineering and Technology
15. A.Kogilavani and Dr.P.Balasubramani, "Clustering and Feature Specific Sentence Extraction Based Summarization of Multiple Documents", International Journal of computer science & information Technology (IJCSIT) Vol.2, No.4, August 2010.