www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 5 May 2015, Page No. 12118-12122

PlSI Is A Way To Summarize The Search Result

Rachana C. Patil[1], Prof. S. R. Durugkar[2]

[1]Savitribai Phule University Pune, S. N. D. College of Engineering & Research Center, Yeola - 423401, India
rachanap4@gmail.com

[2] Savitribai Phule University Pune, S. N. D. College of Engineering & Research Center, Yeola - 423401, India
santoshdurugkar@gmail.com

Abstract: Recently, due to the availability of vast information, which results more difficult to find out and discover what we need. We need tools which help us to establish, examine and recognize these huge quantities of information. For automatically establishing, understanding, examining, and summarizing large automatic accounts, subject modeling delivers some processes: 1. Determine the unseen subjects in the collection 2. Make notes on the documents giving to these subjects 3. Use comments to establish, review and examine. Hence for these purpose Probabilistic Latent Semantic Indexing approach is used to automate document indexing by using a statistical latent class model for factor analysis of count data. In this paper, we find out a set of query-focused which are summarizing from search results. As there may be several subjects related to a given query in the search results, hence to summarize these results in proper order, they should be classified into subjects, and then every subject should be summarized separately. There are two types of redundancies need to be reduced in this summarization process. First, every subject summary should not comprise any redundancy. Second, a subject summary should not be analogous to any other subject summary. In the summarization process, we emphasis on the document grouping process as well as reducing the redundancy between summaries. In this paper, we also suggest the PLSI approach which is a way to summarize the search results. Due to the process of evaluation results, our method accomplishes well in categorizing search results and reducing the redundancy between summaries.

Keywords: Query, summarization process, document clustering, probabilistic latent semantic indexing

1. Introduction

With the beginning of digital records and communication networks, massive sources of word-based data have become accessible to a huge public. Nowadays, the World Wide Web encompasses huge amounts of information. Hence, search engines are essential, if we want to make efficient use of that information. Though, by using the search engines (such as Google, Bing and so on) they usually return only a long list having the title and a snippet of each of the recovered documents. However, these lists are active for directional queries, but they are not useful to users with informational queries. Some systems represent keywords associated to a given query composed with the search results. It is hard for users to recognize the relation between the given keywords and the query, such as the keywords are just words or idioms beyond the context. We report the task of producing a set of query-focused summaries from search results which is representing information about a given query via usual sentences, to solve this problem. Since there are typically many subjects associated to a query in the search results, so to express, multi-subject multi-document summarization by doing the job of summarizing these results. If we do study on multi-document summarization then they usually report summarizing documents associated to a single subject [2]. However when considering the summarization of search results then we want to report summarizing documents which are associated to a several subjects.

When we summarize the documents then they are having various subjects, therefore it is essential to classify them into subjects. For example, if a set of documents related to typhoid then they contains subjects such as the incidences of typhoid, the measures to extravagance typhoid, so on, and the documents should be separated into these subjects and summarized individually. In this process a method for clustering should be active, as one document may be possessed by numerous subjects. In the process of summarization, there are two types of redundancies need to be addressed [2].

First, the summary of each subject should not have any redundancy. We mention to this problem as redundancy inside a summary. In the field of multi-document summarization this problem is well known and there are many methods have been projected to resolve it, such as Maximum Marginal Relevance (MMR) using Integer Linear Programming (ILP) [7][11].

Second, the summary of one subject should not be similar to any of the other subject summaries. We mention to this problem as redundancy among summaries. For example, to summarize the above stated documents associated to typhoid, the summary for occurrences should contain specific information about occurrences, whereas the summary for measures should encompass specific information about measures. This problem is distinguishing from multi-subject multi-document summarization. Some of the methods have been expected to produce subject summaries from documents [1][4].

In this paper, we emphasis on the process of document clustering as well as decrease the redundancy among summaries in the process of summarization. Besides, we recommend a method to summarize search results using PLSI approach. In this method, we work on PLSI to evaluate the association degree of each document to each subject, and then categorize the search results into subjects by using that information. Likewise, in order to decrease the redundancy between summaries, we employ PLSI to evaluate the association degree of each keyword to each subject, and then extract the essential sentences specific to each subject using this information. Due to the process of estimation results, our method achieves well in categorizing search results and reducing the redundancy between summaries [2][3].

2. Our Approach

2.1 Overview

Figure 1 shows an overview of our approach, which includes the following four steps:

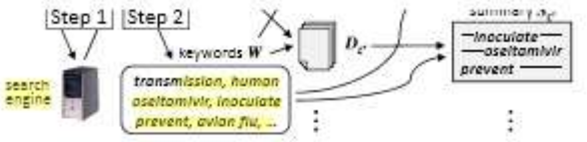


Figure 1: Overview of the system

Step1. Acquisition of Search Results
Obtain the search results for a given query by using a search engine.

Step2. Keyword Extraction
Using the method which is proposed by Shibata et al. (2009) we extract the keywords associated to the query from the search results.

Step3. Document Clustering
With the help of PLSI to estimate the association degree of each document to each subject and then categorizes the search results into subjects.

Step4. Summarization
From each document cluster to generate a summary by extracting the significant sentences specific to each subject.

In the next subsections, we define each step in detail manner.

2.2 Step 1. Acquisition of Search Results

First, by using a search engine (such as Google, Bing and so on), we gain the search results for a given query. To be more accurate, we find the top N^1 documents with the help of search engine results. Next, simple filtering method is used to eliminate those documents that should not be involved in the summarization, such as collections of link. For example, we think any document that has several links as a link collection, and then exclude it. In this paper, after the filtering process the search results are denoted by using *D* and let $N = |D|$.

2.3 Step 2. Keyword Extraction

With the help of method which is proposed by Shibata et al. (2009) we can extract the keywords associated to a query from *D*, which contains the following four steps:

Step2-1. Relevant Sentence Extraction
To extract the sentences which having the query and the sentences nearby the given query as relevant sentences, for each document in *D.*

Step2-2. Keyword Candidate Extraction
Extract composite nouns and parenthetic strings as keyword candidates for each relevant sentence.

Step2-3. Synonymous Candidate Unification
Find the paraphrase pairs and the orthographic variant pairs in the keyword candidates, and merge them.

Step2-4. Keyword Selection
Each keyword candidate should be mark, rank them, and select the best *M* as the keywords associated to the given query.

In this paper, the extracted keywords are denoted by *W*.

2.4 Step 3. Document Clustering

Using Probabilistic Latent Semantic Indexing (PLSI) approach we can classify *D* into subjects. In PLSI, a document *d* and a word *w* are supposed to be provisionally independent given a subject *z*, and the joint probability *p(d,w)* is calculated as given below.

$$p(d,w) = \sum_z p(z)\, p(d\,|\,z)\, p(w\,|\,z) \qquad (1)$$

p(z), p(d|z), and *p(w|z)* are estimated by exploiting the log-likelihood function *L*, which is calculated as follows

$$L = \sum_d \sum_w freq(d,w) logp(d,w), \qquad (2)$$

Where, the frequency of word *w* in document *d* is denoted by *freq(d,w)*. By using the EM algorithm, (in which the E-step and M-step are given below) *L*, is maximized.

E-step

$$\mathrm{p(z|d,w)} = \frac{\mathrm{p(z)\ p(d|z)\ p(w|z)}}{\mathrm{z'\ p(z')\ p(d|z')\ p(w|z')}} \qquad (3)$$

M-step

$$\mathrm{p(z)} = \frac{\sum_d \sum_w \mathrm{freq(d,w)\ p(z|d,w)}}{\sum_d \sum_w \mathrm{freq(d,w)}} \qquad (4)$$

$$\mathrm{p\ (d|z)} = \frac{\sum_w \mathrm{freq(d,w)\ p(z|d,w)}}{\sum_{d'} \sum_w \mathrm{freq(d',w)\ p(z|d',w)}} \qquad (5)$$

$$\mathrm{p(w|z)} = \frac{\sum_d \mathrm{freq(d,w)\ p(z|d,w)}}{\sum_d \sum_{w'} \mathrm{freq(d,w')\ p(z|d,w')}} \qquad (6)$$

The EM algorithm repeats through these steps until merging.

There are the number of subjects K, the search results *D*, and the keywords *W* as input, and estimate *p(z), p(d|z),* and *p(w|z),* where *z* is a subject related to the query, *d* is a document in *D*, and *w* is a keyword in *W*. Though, there is no way of knowing the value of *K*; that is, in advance we do not know how many subjects associated to the query in the search results. Therefore, for numerous values of *K*, we execute PLSI approach and then select the K that has the minimum Akaike Information Criterion (AIC) (Akaike, 1974), calculated as follows [1].

$$\mathrm{AIC = -2L + 2K(N + M)} \qquad (7)$$

Furthermore, we select *p(z), p(d|z),* and *p(w|z)* estimated using the selected *K* as the result of PLSI.

Then, the membership degree of each document to each subject is calculated. The membership degree of document *d* to subject *z*, denoted *p(z|d)*, is calculated as

$$\mathrm{p(z|d)} = \frac{\mathrm{p(d|z)\ p(z)}}{\sum_{z'} \mathrm{p(d|z')}} \qquad (8)$$

Finally, we collect those documents whose membership degree to the subject is larger than the threshold α, for every subject. If there is a document whose membership degree to multiple subjects is larger than the threshold, we classify the document into each subject. In this paper, D_z is used to denote the documents classified into subject *z*.

2.5 Step 4. Summarization

For each subject, we find out the important sentences specific to that subject from each document cluster.

Figure 2: Algorithm for summarization.

```
$Input: A\ set\ of\ K\ document\ clusters \{D_z\}\ (z \in Z)$
$Output: A\ set\ of\ K\ summaries \{S_z\}\ (z \in Z)$
$Procedure:$
1: $for\ all\ z \in Z$
2:   $while |S_z| < num\ (z)$
3:     $for\ all\ s \in D_z$
4:       $calculate\ s\_score(z,\ s,\ S_z)$
5:     $s_{max} = argmax_{s \in Dz \setminus Sz}\ s\_score(z,\ s,\ S_z)$
6:     $S_z = S_z \cup \{s_{max}\}$
7: $return\ S_z$
```

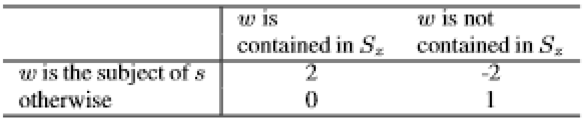
Figure 2 gives the algorithm for summarization. We calculate the importance of sentence *s* to subject *z*, denoted as *s_score(z, s, S_z)*, for each sentence in D_z (lines 3-4), when we produce the summary S_z for subject *z*,. After that we extract the sentence s_{max} with the maximum importance as an important sentence, and include s_{max} in S_z (lines 5-6). Then we recalculate the importance *s_score(z, s, S_z)* for each sentence in D_z except the sentence in S_z (lines 3-4), when we extract the next important sentence. Next we extract the sentence s_{max} with the maximum importance as an essential sentence, and add s_{max} to S_z (lines 5-6). We convey on this process till the number of important sentences combining the summary, which is denoted as $|S_z|$, then reaches the number of important sentences extracted for subject *z*, denoted *num(z)* (line 2).

S_score(z, s, S_z) is calculated as follows:

$$s\_score(z,s,S_z) = \sum_{w \in Ws} (w\_score(z,w) \text{ x } c\_score(w,S_z,s)) \quad (9)$$

Where, W_s represents the keywords in sentences.

Table 1: Values of *c_score(w, S_z, s)*.



	w is contained in S_z	*w* is not contained in S_z
w is the subject of *s*	2	-2
otherwise	0	1

For decreasing the redundancy between summaries, and denotes the importance of keyword *w* to subject *z* we have to use function *w_score(z,w)*. *p(w|z)* is used the probability of *w* given *z*, as the *w_score(z,w)*. If there are keywords with a high probability in both subject *z* and another subject *z*′ then this approach fails, the sentences having such keywords are extracted as the important sentences in both subjects, and it monitors that the generated summaries will have redundancy. Thus, for resolving this problem, we use the association degree of keyword *w* to subject *z*, this is denoted as *p(z|w)*, as *w_score(z,w)*. Using PLSI *p(z)* and *p(w|z)* are estimated in above section 2.4 to compute *p(z|w)*.

$$p(z|w) = \frac{p(w|z)\ p(z)}{\sum_{z'} p(w|z')} \quad (10)$$

The high probability keywords in several subjects should have a low association degree to each subject. Therefore, with the help of *p(z|w)* as the *w_score(z,w)* avoids extracting sentences which having such keywords as important sentences, and they monitors that the similarity among the summaries is decreased. Also, the keywords which are specific to a subject are supposed to have a high association degree to that subject. Therefore, to extract sentences including such keywords as important sentences easily by using *p(z|w)* as *w_score(z,w)*, and each summary is specific to the exact subject is indicated with the result.

The function of *c_score(w, s_z, s)* is used to decrease the redundancy within a summary, and it also signifies the importance of a keyword *w* in a sentence *s* under the situation that there is a number of extracted important sentences S_z. If whether or not *w* is contained in S_z, then the value of *c_score(w, s_z, s)* function is determined. Table 1 shows the values of *c_score(w, s_z, s)*. We set *c_score(w, s_z, s)* = 0, if *w* is contained in S_z, , else we set *c_score(w, s_z, s)*= 1. Like this, we can find out the sentences which having the keywords that are not contained in S_z as important sentences, and decrease the redundancy inside the summary. For instance, we set *c_score(w, s_z, s)*= 2, even if w is contained in S_z, as long as w is the subject of *s*. Similarly, we set *c_score(w, s_z, s)*= −2, even if w is not contained in S_z, as long as w is the subject of *s*. These values for *c_score(w, s_z, s)*are determined properly.

Lastly, with the use of *p(z)* function we determine the set of significant sentences extracted for subject *z*, which is denoted as *num(z)*.

$$num(z)= \lfloor I \times p(z) \rfloor (p(z) \geq \beta) \quad (11)$$

Where, *I* represent the parameter that controls the total number of important sentences extracted for each subject.

3. Experimental Results

3.1 Overview

According to our method, we prepared a system, and requested the subjects to use our system to estimate the following four features of our method.

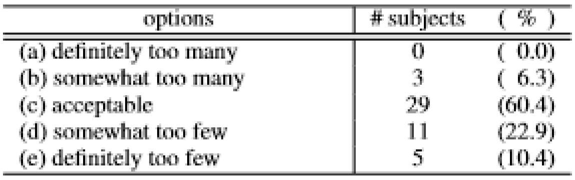
- Validity of the set of subjects
- Accuracy of document clustering
- Degree of decrease in redundancy among summaries
- Efficiency of the method for presenting information through summaries

We permitted the subjects to create random queries for this system.

3.2 Validity of the set of subjects

First, we examined how well the projected method determined the group of subjects. In that method, by using AIC the number is determined. Rather, in the search results we should have physically calculated the subjects, and then using AIC we compared this with the number which is to be determined. If the search results contained 1,000 documents then it was difficult to count the subjects. Moreover, for each query given by each subject was impossible to count the number of subjects. Hence, in this examination, we simply queried the subjects whether they handled the set of subject summaries represented to them was appropriate or not, and in terms of usability examined our method. Table 2 gives the results. According to those results, it looks that users are satisfied with the system presenting about 3 or 4 subject summaries, and in terms of usability our method determined the desirable set of subjects.

Table 2: Validity of the set of subjects.



options	# subjects	(%)
(a) definitely too many	0	(0.0)
(b) somewhat too many	3	(6.3)
(c) acceptable	29	(60.4)
(d) somewhat too few	11	(22.9)
(e) definitely too few	5	(10.4)

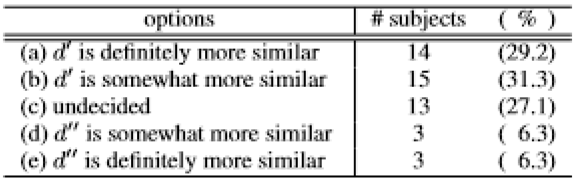
3.3 Accuracy of document clustering

Second, we examined how exactly the new method classified the search results into subjects. To be more detailed, we estimated the consistency of the association degree *p(z|d)* used in the process of document clustering. It is usually hard to evaluate clustering methods. In this case, we did not have any accurate data and could not even create these subsequently, as per mentioned earlier, the set of subjects is not recognized. Similarly, it is not possible to categorize by hand search results which having 1,000 documents. Thus, by comparing correct data with the clustering result from our method, we did not estimate this method directly, but instead estimated it indirectly by examining the consistency of the association degree *p(z|d)* used in the process of document clustering.

The evaluation process is as given below. First, we represented the subjects with a document d, which was projected by this system to have a high association degree to a subject *z*. We selected as *d*, a document with a association degree of about 0.9. Next, we presented two documents to the subjects. One was a document *d′* whose association degree to *z* was also about 0.9, and other was a document d″ whose association degree to *z* was about 0.1. Finally, we questioned them which document was more similar.

Table 3 gives the results. From these results we recognize subjects in our system is in arrangement to some amount with the subject that is, this method was able to estimate a reliable association degree *p(z|d).* Hence, it seems that this method using *p(z|d)* function is able to categorize search results into subjects to some extent.

Table 3: Accuracy of the estimation *p(z|d).*



options	# subjects	(%)
(a) d' is definitely more similar	14	(29.2)
(b) d' is somewhat more similar	15	(31.3)
(c) undecided	13	(27.1)
(d) d'' is somewhat more similar	3	(6.3)
(e) d'' is definitely more similar	3	(6.3)

3.4 Degree of decrease in redundancy among summaries

Third, we examined how well the new method decreased the redundancy among summaries. To be more specific, we used three measures as *w_score(z,w)* to produce summaries and examined which measure generated the least redundant summaries. Usually, methods for decresing redundancy are estimated using ROUGE (Lin, 2004), BE (Hovy et al., 2005), or Pyramid (Nenkova and Passonneau, 2004). However, the use of these methods requires that model summaries are produced by humans, and this was not probable for the same reason as mentioned earlier. Hence, we did not execute a direct estimate using the methods such as ROUGE, but in its place estimated how well this method executed in decreasing redundancy between summaries using the association degree *p(z|w)* as *w_score(z,w),* [5][6][12].

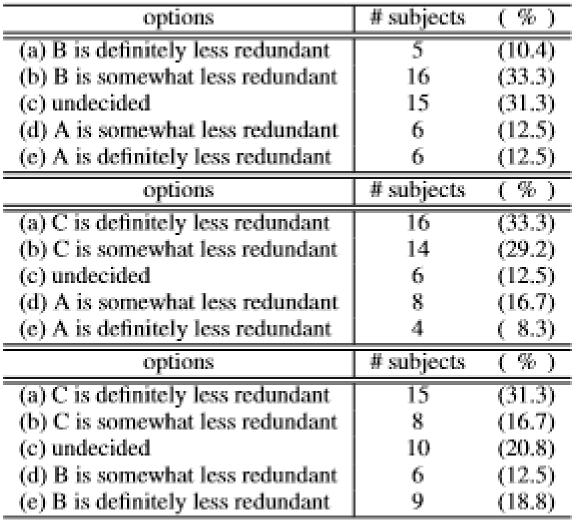
The evaluation process was as follows. We used three measures as *w_score(z,w),* and generated three sets of summaries [10].

Summaries A This set of summaries was generated using *dfidf(w)* as *w_score(z,w),* with *dfidf(w)* calculated as *ldf(w)×log(100million/gdf(w)),ldf(w)* presenting the document frequency of keyword w in the search results, and *gdf(w)* representing the document frequency of keyword *w*.

Summaries B Using *p(w|z)* as *w_score(z,w),* this set of summaries was generated.

Summaries C By using p(z|w) as w_score(z,w), this set of summaries was produced. Then we represented the subjects with three pairs of summaries, namely a pair from A and B, a pair from A and C, and a pair from B and C, are queried them which summaries in each pair was less redundant4. The results are given in Tables 4.

Table 4: Comparison of *dfidf(w*), *p(w|z)* and *p(z|w).*



options	# subjects	(%)
(a) B is definitely less redundant	5	(10.4)
(b) B is somewhat less redundant	16	(33.3)
(c) undecided	15	(31.3)
(d) A is somewhat less redundant	6	(12.5)
(e) A is definitely less redundant	6	(12.5)
options	# subjects	(%)
(a) C is definitely less redundant	16	(33.3)
(b) C is somewhat less redundant	14	(29.2)
(c) undecided	6	(12.5)
(d) A is somewhat less redundant	8	(16.7)
(e) A is definitely less redundant	4	(8.3)
options	# subjects	(%)
(a) C is definitely less redundant	15	(31.3)
(b) C is somewhat less redundant	8	(16.7)
(c) undecided	10	(20.8)
(d) B is somewhat less redundant	6	(12.5)
(e) B is definitely less redundant	9	(18.8)

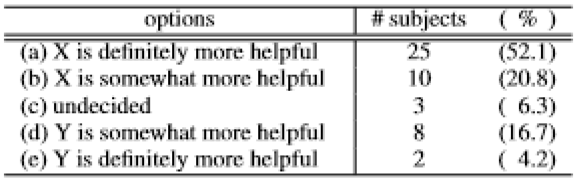
3.5 Efficiency of the method for presenting information through summaries

We also examined the efficiency of the method for representing information through summaries. We queried the subjects to match two different ways of representing information and to judge which way was most active in terms of helpfulness for gathering information about a given query.

One of the methods represented the search results with subject summaries produced by our system (method A), and while the other method represented the search results with the keywords involved in each subject summary (method B).

Table 5 gives the results. From these results, it appears that the method of representing information through summaries is effective in terms of helpfulness for gathering information about a given query.

Table5: Comparison of summaries and keywords



options	# subjects	(%)
(a) X is definitely more helpful	25	(52.1)
(b) X is somewhat more helpful	10	(20.8)
(c) undecided	3	(6.3)
(d) Y is somewhat more helpful	8	(16.7)
(e) Y is definitely more helpful	2	(4.2)

4. Conclusion

In this paper, we focused on the task of producing a set of query-focused summaries from search results. A process of categorizing them into subjects associated to the query was required to summarize the search results for a given query. In this method, we used PLSI to estimate the association degree of each document to each subject, and then categorized search results into subjects. The evaluation results displayed that our method projected consistent degrees of association. Hence, it seems that our method is capable to some extent to categorize search results into subjects.

In the process of summarization, redundancy within a summary and redundancy between summaries needs to be decreased. The PLSI approach is used in our method to evaluate the association degree of each keyword to each subject, and then extracted the important sentences specific to each subject. The estimation results displayed that our method was capable to decrease the redundancy between summaries using the association degree.

5. References

[1] Akaike, Hirotugu. 1974. A new look at the statistical model identification. IEEE Transactions on Automation Control, 19(6):716–723. J. Cui, F. Wen, and X. Tang, Real Time Google and Live Image Search Re-Ranking, Proc. 16th ACM Intl Conf. Multimedia, 2008.

[2] Jun Harashima and Sadao Kurohashi. Summarizing Search Results using PLSI. In Proceedings of the Second Workshop on NLP Challenges in the Information Explosion Era (NLPIX 2010).

[3] Filatova, Elenaand Vasileios Hatzivassiloglou. 2004. A formal model for information selection in multi sentence text extraction. In Proceedings of COLING 2004.

[4] Haghighi, Aria and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In Proceedings of HLT-NAACL 2009.

[5] Hovy, Eduard, Chin-Yew Lin, and Liang Zhou. 2005. Evaluating duc 2005 using basic elements. In Proceedings of DUC 2005.

[6] Lin, Chin-Yew. 2004. Rouge: A package for automatic evaluation of summaries. In Proceedings of ACL 2004 Workshop on Text Summarization Branches Out.

[7] McDonald, Ryan. 2007. A study of global inference algorithms in multi-document summarization. In Proceedings of ECIR 2007.

[8] Shibata, Tomohide, Yasuo Bamba, Keiji Shinzato, and Sadao Kurohashi. 2009. Web information organization using keyword distillation based clustering. In Proceedings of WI 2009.

[9] Shinzato, Keiji, Daisuke Kawahara, Chikara Hashimoto, and Sadao Kurohashi. 2008a. A large-scale web data collection as a natural language processing infrastructure. In Proceedings of LREC 2008.

[10] Shinzato, Keiji, Tomohide Shibata, Daisuke Kawahara, Chikara Hashimoto, and Sadao Kurohashi. 2008b. TSUBAKI: An Open Search Engine Infrastructure for Developing New Information Access Methodology. In Proceedings of IJCNLP 2008.

[11] Takamura, Hiroya and Manabu Okumura. 2009. Text summarization model based on maximum coverage problem and its variant. In Proceedings of EACL 2009.

[12] Nenkova, Ani and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In Proceedings of NAACL-HLT 2004

Author Profile

Ms. Rachana C. Patil received the B.E. degree in Computer Engineering from KCESCOEIT, Jalgaon, in 2011. She is now pursuing her M.E. from S. N. D. College of Engineering & Research Center, Yeola.