AUTOMATIC TEXT SUMMARIZATION USING SUPERVISED MACHINE LEARNING TECHNIQUE FOR HINDI LANGAUGE

Nikita Desai[1], Prachi Shah[2]

[1]*Associate Professor, IT Department ,Dharmsinh Desai University, Gujarat, India*
[2]*Student, IT Department, Dharmsinh Desai University, Gujarat, India*

Abstract

Automatic text summarization is a technique which compresses large text into a shorter text which includes the important information. Hindi is the top-most language used in India and also in a few neighboring countries there is a lack of proper summarization system for Hindi text. Hence,in this paper, we present an approach to the design an automatic text summarizer for Hindi text that generates a summary by extracting sentences. It deals with a single document summarization based on machine learning approach. Each sentence in the document is represented by a set of various features namely- sentence paragraph position, sentence overall position, numeric data, presence of inverted commas, sentence length and keywords in sentences. The sentences are classified into one of four classes namely- most important, important, less important and not important. The classes are in turn having ranks from 4 to 1 respectively with "4"indicating most important sentence and "1" being least relevant sentence . Next a supervised machine learning tool SVM^{rank} is used to train the summarizer to extract important sentences, based on the feature vector. The sentences are ordered according to the ranking of classes. Then based on the required compression ratio, sentences are included in the final summary. The experiment was performed on news articles of different category such as bollywood, politics and sports. The performance of the technique is compared with the human generated summaries. The average result of experiments indicates 72% accuracy at 50% compression ratio and 60% accuracy at 25% compression ratio.

Keywords: *Hindi Text Summarization; Supervised Machine Learning; SVM; Text Mining; Sentence Extraction; Summary Generation*

----------***----------

1. INTRODUCTION

The rapid growth of the Internet has resulted in enormous amount of information in natural language that has become increasingly more difficult to access efficiently. In the fast moving world it's difficult to read all the text-content. The technique is being developed through research since last 50 years and with the increased use of the Internet and electronic documents the need for fast and reliable summarization has undergone a rapid growth. Automatic text summarization is a technique which compresses large text to a shorter text which includes the important information, subjects and various elements from the original text. The computer is given a text and returns a summary of the original text. Text summarization can be useful in day to day life. For instance, review of a movie, headlines of news, abstract summary of technical paper and review of book. Also, it is more and more being used in the commercial sector such as, data mining of text databases, web-based information retrieval, telephone communication industry, in word processing tools, etc. Automatic text summarization is one the important steps for information managing tasks. The problem of selecting the most important parts of the text is solved by it.

The output of summary can be of two types: Extractive summaries and abstractive summaries. Extractive summaries are produced by extracting the whole sentences from the source text. The importance of sentences is determined based on statistical and linguistic features of sentences. Abstractive summaries are produced by reformulating sentences of the source text. An Abstractive summarizers [12] [13] understands the main concepts in a document and then convey those concepts in clear natural language. It uses linguistic methods to examine and interpret the text and then to find the new concepts and terms to best describe it by generating new shorter text that conveys the most significant information from the original text document [5].

Although the various approaches are proposed for summarization of major languages like English, Swedish, etc. some challenging problems are still open for other languages of the world. India has around 120 major languages and 1500 other languages. Since recent years many researchers are working on Indian and other languages but less work has done on Indian languages. Though Hindi is the top-most language used in India and also in a few neighboring countries there is a lack of proper summarization system for Hindi text. Hence, the technique for Hindi text summarization has been proposed in this paper.

The rest of the paper is structured as follows: Section 2 describes the related work in text summarization, and Section 3 describes the proposed summarization technique. Section 4 presents the experimental work and its results and Section 5 concludes the paper by summarizing the study and gives some future work.

2. THE RELATED WORK

A lot of a variety of summarization methods has been proposed and evaluated, but mostly for English and European languages. The very first work in automated text summarization was done by Luhn [1] in 1958. He used frequency of word occurrence to produce summaries from technical documents with the aim to determine the relevance of a sentence in a document. It has been assumed that the most frequent words are indicative of the main topic of a document. P.B. Baxendale [2] in 1958 proposed novel feature that is sentence location or sentence location position in an input document. It is analyzed that sentences which are positioned at the beginning or at the end of the document are more important than other sentences in the document. A sentence position feature became an essential feature for sentence extraction and it is used till now. H.P. Edmundson [3] in 1969 proposed a novel structure for a text summarization. He proposed two new features. First, cue-words that is presence of most indicative words into a document such as finally, in summary, lastly, etc. Second, title or heading Words that is an additional weight was assigned to a sentence, if sentences have heading words in it.

Later, Julian Kupiec [4] in 1995 proposed some new features like sentence length, presence of uppercase words, phrase structure and also included other features developed previously. Also, described a new technique of summarization with naive-Bayes classifier, the classification function classifies each sentence as whether it is extraction worthy or not.

Chin-Yew Lin [6] in 1999 also tried to model system for sentence extraction using decision trees for creating informative generic/query-oriented extracts. Chin-Yew Lin identified different features such as: title, baseline, position, tf-idf, query signature, sentence length, lexical connectivity, quotation, first sentences, proper noun, pronoun and adjective, numerical data, weekdays and month. The score for all features were combined by the automatic learning using decision trees and combination function. Lin conducted deep study of different features effect through glass-box. The experimental result shows that there was no single feature which performs well for query-based summaries. Features like numerical data gives better results for query which requires answer in numerical value, while weekdays and month feature gives best result for queries like "when?"

Conroy and O'leary [7] created a system for sentence extraction using two techniques named QR and HMM. In QR technique the importance of sentence was calculated and the sentences having higher importance were added to the summary. Then, relative score of remaining sentences were altered because some of the remaining sentences were redundant. And the other technique was hidden markov model (HMM) that is sequential model for automatic text summarization. And in HMM only three features were used: sentence position, total no of terms in the sentence and similarity of sentence terms in the given document. In the end, evaluation was done by comparing HMM generated summary with human generated summary.

S. P. Yong et al [8] in 2005 described an automated system that had ability of learning by combining different recent approaches such as: statistical approach, sentences extraction and neural network.

Ladda Suanmali et al [9] in 2009 proposed a system based on fuzzy logic. Fuzzy rules and fuzzy sets were used for extracting significant sentences based on sentence features. Various features were used to compute sentence significance and a value was given between '0' and '1' if a particular feature was present in the sentence. Feature used were title word in sentence, sentence centrality, similarity to first sentence, keyword, sentence length, sentence position, proper noun and numeric data. The values of these features were given as input to fuzzy system, which created an output based on features and IF-THEN rules defined for sentence extraction.

Li Chengcheng [10] in 2010 proposed a novel ATS method based upon Rhetorical Structure Theory (RST). RST is the notations of rhetorical relation presented between two non-overlapping text called nucleus (N) and satellite (S). The experimental observations were used to express difference between N and S, which was expressed by N that what is more important to the writer's purpose than S and the Rhetorical nucleus was comprehensible independent of the S. The entire text was divided into small units called sentences based on delimiter like full stop, commas, any punctuation mark found between sentences. The entire process was built on a graph, sentences which were less important were removed from the graph and remaining sentences were summarized. More detailed survey and comparisons of related work in text summarization field can be found in [16] .

3. PROPOSED METHOD

The goal of automatic text summarization is to select the most important sentences of the Hindi text document. The proposed method uses various statistical features to find most important sentences. Also it makes use of SVM^{rank} a machine learning tool to rank the sentences. The general outline of the methodology used for this task is described below.

Input: A text file (original -Og).

Output: A summarized text(S) of original (Og), as per compression ratio.

1. Read Input text File -Og
2. Pre-process the file Og . *//Preprocessing step*
2.1 Segment text file into sentences.
2.2 Tokenize each sentence into words.
2.3 Remove stop-words.

3. *//Processing step*
3.1 Extract following features from Og file -
Sentence Paragraph Position (f1),

Sentence Overall Position (f2),
Numerical Data in Sentence (f3),
Presence of Inverted Commas (f4),
Sentence Length (f5),
Keywords in Sentence (f6)

3.2 Apply SVM model to rank sentences in range from4 to 1,with "4" indicating most important sentence and "1" indicating not important sentence.

4. Generate summary . *//Extraction step*

While (lines in summary file (S) does not exceed maximum limit as per given by compression ratio) Do

4.1Extract all lines from Og with rank4
4.2 Extract all lines from Og with rank3
4.3 Extract all lines from Og with rank2

5. Display summary file (S).

Thus the proposed technique can be grouped into 3 major blocks- Pre-processing, Processing and Extraction which are explained henceforth.

3.1 Pre-Processing

In pre processing stage of this proposed technique, the text is first split into sentences, then sentences are further broken into words and then stop words are removed. Preprocessing stage involves 3 steps 1) Segmentation 2) Tokenization 3) Stop words removal.

3.1.1 Segmentation

In this segmentation part, sentences are segmented based on sentence boundary. In Hindi language sentence boundary is recognized by "|". On every sentence boundary, the sentence are broken and put into list. The output of sentence segmentation part is collection of sentences that are further processed in next stages.

3.1.2 Tokenization

In this tokenization part, sentences are broken down into words. In Hindi, sentences are tokenized by identifying the space and comma between the words. So the list is created in which it contains elements as words or also called tokens which are maintained for further processing.

3.1.3 Stop-Words Removal

Most commonly or frequently used words are called stop-words. Stop-words are worthless and do not have any importance into the sentences. So these types of words should be removed from input text document, otherwise the sentence having more no of stop-words could have higher weight. We analyzed that every Hindi text document contains minimum 25-30% or more stop-words. Example of stop-words are "के", "है", "और", "नहीं" etc.

The input and corresponding sample output of the preprocessing phase is shown in fig1 and fig2 respectively.

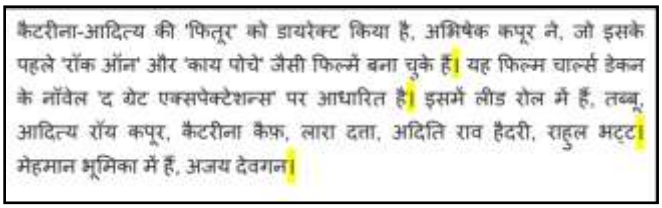
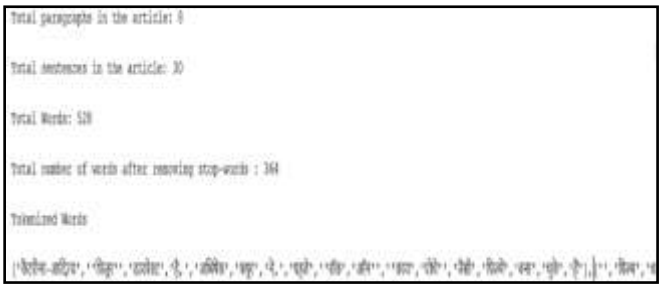

कैटरीना-आदित्य की 'फितूर' को डायरेक्ट किया है, अभिषेक कपूर ने, जो इसके पहले 'रॉक ऑन' और 'काय पोचे' जैसी फिल्में बना चुके हैं। यह फिल्म चार्ल्स डेकन के नॉवेल 'द ग्रेट एक्सपेक्टेशन्स' पर आधारित है। इसमें लीड रोल में हैं, तब्बू, आदित्य रॉय कपूर, कैटरीना कैफ, लारा दत्ता, अदिति राव हैदरी, राहुल भट्ट। मेहमान भूमिका में हैं, अजय देवगन।

Fig1. Sample input


Total paragraphs in the article: 8

Total sentences in the article: 30

Total Words: 519

Total number of words after removing stop-words : 364

Tokenized Words

Fig. 2 Sample output after preprocessing phase

3.2 Processing

Processing stage is the heart of text summarization, here in depth analysis on text document is done. In processing stage, value of feature for every sentence is calculated. Our proposed technique makes use of six statistical features [15] for calculating sentence score and each of them are explain below:

3.2.1 Sentence Paragraph Position (f1)

Position of sentence has its importance in a text document. Starting sentences of paragraphs are important in nearly all the cases because they convey theme of the document and has higher probability to be extracted for the summary. Sentence position value is calculated in such a way that, higher values are assigned to the starting sentences and lower values are assigned to ending sentences of the paragraph. Sentence Paragraph Position values are calculated using the equation given in Eqn.1 :

$$Sen\_para\_pos(Si) = \frac{n - i}{n}$$

Eqn.1

In Eqn 1,*Si* is the i^{th}sentence in a paragraph, n= Total no of sentences in a paragraph and "i" ranges from 0 to n.

3.2.2 Sentence Overall Position (f2)

Sentence Overall Position values are calculated in context of the entire text in the document. Sentence position value is calculated in such a way that, higher values are assigned to the starting sentences and lower values are assigned to ending sentences of the document. Sentence Overall Position values are calculated using the equation given in Eqn.2:

$$Sen\_overall\_pos(Si) = \frac{n - i}{n}$$

Eqn.2

In eqn2, Si is the i^{th} sentence in a text document, n= Total no of sentences in a text document and i ranges from 0 to n.

3.2.3 Numerical Data in Sentence (f3)

A numeric data in Hindi Sentence represent some important information regarding some date, age, rupees, address etc. In Hindi, data can be represented in digits and Hindi fonts such as (01/01/2014 // ०१/०१/२०१४) date, etc. Numeric data values are calculated using the equation given in Eqn.3.

$$Num\_Data(Si) = \frac{(\text{numeric data in (Si)})}{\text{SenLen(Si)}}$$

Eqn.3

Where numeric data in (*Si*) = Total no of numeric data in sentence *Si* and SenLen(*Si*)= Total no of words in sentence(*Si*).

3.2.4 Presence of Inverted Commas (f4)

In Hindi , text having quotation marks or inverted comma surrounding it indicates direct speech, literal title or name etc. and it contains important information. Presence of inverted commas is calculated using equation given in Eqn.4 below:

$$IC(Si) = \frac{(\text{Quoted Words in (Si)})}{\text{SenLen(Si)}}$$

Eqn.4

Where QuotedWords in (Si) = Total no of (Si) words between quotation marks**.**

3.2.5 Sentence Length (f5)

Sentences which are shorter in length may not represent theme of a text document because of fewer words contained in it, although selecting longer length sentences are also not good for summary. So sentence length values are calculated in such a way that, shorter and longer sentences are assigned lower values. Sentence length values are calculated using equation given in Eqn.5:

$$Sen\_len(Si) = \frac{\text{Words in (Si))}}{\text{LongSen}}$$

Eqn.5

Where Words (*Si*)= Total no of words in sentence *Si* and LongSen= Total no of words in the longest sentence.

3.2.6Keywords in Sentence (f6)

Keywords are words that appear with high frequency in a text document. Keywords identification and computation is very important feature and it helps in deciding sentence importance. Keywords in sentences are calculated using equation given in Eqn.6:

$$Key(Si) = \frac{\text{Keywords(Si)}}{\text{SenLen(Si)}}$$

Eqn.6

Where key(*Si*) = Total no of keywords in sentence *Si*.

3.3 Extraction

The sentences are now extracted and included in the final summary file based on the total lines possible in depending on the compression ratio intended.First, all the sentences having rank 4 are included in summary. Subsequently sentences having rank 3 and 2 are included in summary. It is noteworthy that the relative position of sentences is not changed in the final summary file .For example if two sentences s1 and s2 were at first paragraph and third paragraph in original text , in summary file alsothe position of s1 would be before s2 sentence .

4. EXPERIMENTS

In this section we discuss about various experiments conducted for generating summary of the Hindi text. For implementing the above discussed approach, Java version 1.8.0_66 as programming language and Netbeans IDE 8.1 as platform have been used as they support UTF-8 format, which is necessary for processing Hindi text. Further, SVM^{RANK} has been used as a machine learning tool to extract important sentences based on the given rank.

4.1 Data Set

As our main focus is on Hindi news domain, we have collected data set from various online news sources like zeenews, khabarNDTV, Patrika, etc. We have collected total 130 Hindi news articles from different categories of news namely bollywood, politics and sports.

Table1. Characteristics of dataset

No. of articles collected	130 Hindi news aricles
Domain categories	Bollywood, Politics, Sports
Average Sentences	20-30
Average Words	400- 500

To generate the tagged corpus , the sentences are classified manually into one of four “ranked” classes namely- most important, important, less important and not important. These classes are ranked from “4” to “1” respectively.

4.2 Evaluation

It is required to check accuracy, relevance and usefulness of the summary created by an automatic summarizer. For evaluating results of summarization, accuracy has been used as evaluation parameters. The generated summaries are evaluated against human generated summaries. The experts were asked to create summaries of our collected dataset.

Accuracy is the percentage from which the sentences are correctly classified for the inclusion in the summary [14].

$$Accuracy() = \frac{no.\ of\ sentences\ correctly\ classified}{Total\ no.\ of\ sentences \in Summary} * 100$$

4.3 Experimental Setup with Results

4.3.1Experiment setup 1

In this experiment, we have used various combinations of features to identify the importance of each feature and tested on Bollywood data set. Total 40 documents are given as training set and results have been tested on 10 documents for various combination of features set.

Table2.Results of Experiment setup 1

Feature set	Features	Average Accuracy
set 1	f1	55%
set 2	f1,f2	67%
set 3	f1, f2, f3	67%
set 4	f1,f2, f3, f4	67%
set 5	f1, f5, f6	59%
set 6	f3, f4, f5, f6	62%
set 7	f2, f3, f4, f5, f6	71%
set 8	f1, f3, f4, f5, f6	61%
set 9	f1, f2, f3, f4, f5	69%
set 10	f1, f2, f3, f4, f5, f6	**72%**

From the results of Table 2, it is evident that considering only sentence paragraph position (f1) feature gives result 55%. While adding sentence overall position (f2) in set 2 increases the results to 67%. Next, including features numeric data (f3) and presence of inverted comma (f4) doesn't make difference in the results as it can be visible in set 3 and 4. Now, by adding features sentence length (f5) and keywords in sentence (f6) with sentence paragraph position (f1) does not make much difference. Now, by removing sentence position features degrades the results. Hence, we can conclude that by considering all the six features results better. It has been now clear that sentence paragraph position (f1), sentence overall position (f2) and keywords in sentence (f6) are considered to be the most important features.

4.3.2 Experiment Seupt2

Next , ten documents from each category have been tested using all the six features at 50% and 25% compression ratio (CR) and the results obtained are tabulated below in 3,4, and 5 for Bollywood ,politics and sports categories respectively.

Table 3. Results on Bollywood Dataset

		50% Compression Ratio			25% Compression Ratio		
Doc ID	**Total No. of Sentences**	**Sentences in Summary**	**Correctly Classified Sentence**	**Accuracy**	**Sentences in Summary**	**Correctly Classified Sentence**	**Accuracy**
1	16	8	6	75%	4	2	50%
2	29	15	11	73%	7	4	57%
3	16	8	6	75%	4	3	75%
4	27	14	11	79%	7	5	72%
5	14	7	5	72%	4	3	75%
6	30	15	11	73%	8	5	63%
7	16	8	5	63%	4	3	75%
8	23	12	9	75%	6	5	83%
9	23	12	8	67%	6	4	67%
10	18	9	5	56%	5	2	40%
Average				**71%**			**66%**

Table4. Results on Politics Dataset

		50% Compression Ratio			25% Compression Ratio		
Doc ID	**Total No. of Sentences**	**Sentences in Summary**	**Correctly Classified Sentence**	**Accuracy**	**Sentences in Summary**	**Correctly Classified Sentence**	**Accuracy**
1	21	11	8	73%	5	4	80%
2	26	13	10	77%	7	4	57%
3	20	10	7	70%	5	4	80%
4	41	20	15	75%	10	6	60%
5	34	17	13	77%	9	5	56%
6	19	10	7	70%	5	3	60%
7	35	17	12	71%	9	7	78%
8	26	13	10	77%	7	5	72%
9	25	13	10	77%	6	3	50%
10	26	13	10	77%	7	3	43%
Average				**74%**			**64%**

Table 5. Results of Sports Dataset

		50% Compression Ratio			25% Compression Ratio		
Doc ID	**Total No. of Sentences**	**Sentences in Summary**	**Correctly Classified Sentence**	**Accuracy**	**Sentences in Summary**	**Correctly Classified Sentence**	**Accuracy**
1	22	11	7	64%	6	2	33%
2	24	12	9	75%	6	2	33%
3	19	10	7	70%	5	2	40%
4	24	12	9	75%	6	3	50%
5	26	13	11	85%	7	5	72%
6	22	11	8	73%	6	4	67%
7	25	13	9	69%	6	3	50%
8	17	8	4	50%	4	2	50%
9	22	11	8	73%	6	2	33%
10	15	8	6	75%	4	3	75%
Average				**71%**			**50%**

Table6. Comparative Results of Expreiment setup 2

Sr No.	Domain	50% compression ratio	25% compression ratio
1	Bollywood	71	66
2	Politics	74	63
3	sports	71	50

It can be observed from the results of Table6 that as the compression ratio decreases the accuracy also decreases. Another interesting observation is as the size of input document increases,the accuracy at 50% compression improves as compared for the small sized input.

4. CONCLUSION

This paper discusses single document automatic text summarization for Hindi text using machine learning technique. As expected it has become clear from the experimental results, the performance in each of the subtasks directly affects the ability to generate high quality summaries. Also it is noteworthy that the summarization is more difficult if we need more compression.

In future, more features like named entity recognition, cue words, context information, world knowledge etc, can be added to improvise the technique. Also, same technique can be applied on domains other than news and later we can study the effects of various domain characteristics on the suggested features and overall performance of the technique. It can also be extended to work on multiple documents. Also it would be interesting to find other suitable machine learning classifiers other than SVM for the task.

ACKNOWLEDGMENT

We would like to thank Mrs. Usha Parikhand Mr. Aashish Shah, expert Hindi language teachers, for donating their valuable time for helping in evaluation of summaries.

REFERENCES

[1]. Hans Peter Luhn. “The automatic creation of literature abstracts,” IBM Journal of research and development, 2(2):159–165, 1958.

[2]. P. B. Baxendale, “Machine-made Index for Technical Literature -An Experiment,” Journal of Research and IBM Development, vol. 2, no. 4, pp. 354-361, October 1958.

[3]. Harold P Edmundson. “New methods in automatic extracting,” Journal of the ACM (JACM), 16(2):264–285, 1969.

[4]. Julian Kupiec, Jan Pedersen and Francine Chen, “A trainable document summarizer,” Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp. 68-73, 1995.

[5]. Gupta Vishal and Gurpreet Singh Lehal. “A survey of text summarization extractive techniques.” Journal of Emerging Technologies in Web Intelligence, vol. 3, pp.258-268,2010.

[6]. Chin-Yew Lin, “Training a selection function for extraction,” Proceedings of the eighth international conference on Information and knowledge management, ACM, pp. 55-62, 1999.

[7]. M.J Conroy and O'leary, “Text summarization via hidden markov models,” In Proceedings of SIGIR '01, pp. 406-407, 2001.

[8]. Md Haque, Suraiya Pervin and others, “Literature Review of Automatic Single Document Text Summarization Using NLP,” International Journal of Innovation and Applied Studies, vol.3, no.3, pp. 857-865, 2013.

[9]. Ladda Suanmali, Mohammed Salem Binwahlan and Naomie Salim, “Sentence features fusion for text summarization using fuzzy logic,” Ninth International Conference on Hybrid Intelligent Systems, IEEE, vol.1, pp.142-146, 2009.

[10]. Li Chengcheng, "Automatic Text Summarization Based On Rhetorical Structure Theory," International Conference on Computer Application and System Modeling (ICCASM), vol. 13, pp. 595-598, October 2010.
[11]. T. Joachims, "Training Linear SVMs in Linear Time", Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD), 2006.
[12]. G Erkan and Dragomir R. Radev, "LexRank: Graph-based Centrality as Salience in Text Summarization", Journal of Artificial Intelligence Research, Re-search, Vol. 22, pp.457-479,2004.
[13]. Udo Hahn and Martin Romacker, "The SYNDIKATE text Knowledge base generator", Proceedings of the first International conference on Human language technology research, Association for Computational Linguistics, ACM, Morristown, NJ, USA, 2001.
[14]. Chuang, Wesley T., and Jihoon Yang. "Extracting sentence segments for text summarization: a machine learning approach." In Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 152-159, 2000.
[15]. Aristoteles, Widiarti, and Eko Dwi Wibowo. "Text Feature Weighting for Summarization of Documents Bahasa Indonesia by Using Binary Logistic Regression Algorithm." International Journal of Computer Science and Telecommunications, vol. 5, no. 7, 2014.
[16]. Prachi Shah and Nikita P.Desai ,"A survey of automatic text summarization tecniques for indian and foreign langauges", International conference on electrical, electronics and optimization tecniques (ICEEOT)-2016.

BIOGRAPHIES

Nikita Desai is Associate professor in IT dept since July 2008. She has total 12+ years of teaching experience. Her major expertise is in Compiler construction, Natural language processing and algorithm analysis.

Ms Prachi Shah was a post graduate student at Masters program in Information Technology department of Dharmsinh Desai university. She received her MTech degree in May, 2016 .