Statistical and analytical study of guided abstractive text summarization

Jagadish S. Kallimani[1,*], K. G. Srinivasa[2] and B. Eswara Reddy[3]

[1]Department of Computer Science and Engineering,
Jawaharlal Nehru Technological University, Kakinada 533 003, India
[2]Department of Computer Science and Engineering,
M. S. Ramaiah Institute of Technology, Bengaluru 560 086, India
[3]JNTUA College of Engineering,
Jawaharlal Nehru Technological University, Anantapur 515 002, India

The process of creating condensed version of given text document by collecting only the important information in it is called abstractive summarization. This involves structuring the information into sentences which are simple and easy to understand. This communication presents the analytical study of the process that generates abstractive summary using unified model with attribute based information extraction (IE) rules and class based templates. Classification of the document into several categories is achieved by term frequency/inverse document frequency (TF/IDF) rules. To generate the information intensive summaries, we use templates for sentence generation. The IE rules are designed to address the complexities involved in Indian regional languages. This paper statistically analyses the adaptation of the methodology over multiple Indian languages and many document categories. Comparisons between abstractive and extractive summaries are also presented.

Keywords: Abstractive and extractive text summarizations, information extraction, language parsing and understanding, template selection, template-based generation.

IN spite of the rapid progress in Natural Language Processing (NLP) techniques, the abstractive summarization methods are tender and persisting research topic. The current abstractive summarization techniques in English language are not comprehensive due to drawbacks in semantic representation, inference and natural language generation[1]. Research in abstractive summarization methodologies for Indian regional languages has started recently. Due to lack of linguistic tools in abstractive summarization, it became more challenging.

Earlier, word and phrase frequency[2], position in the text[3], and key phrases[4] are the features considered in summarizing Indian language documents. Extractive summarization does not combine concepts mentioned in the source document. Rewriting techniques (such as sentence compression, sentence fusion[5]) based on syntactic analysis are explored in abstractive summarization. Topic based guided summarization technique to generate abstractive summary is presented in this paper for few Indian regional languages.

There is an increasing need for automatic summarizers in the context of data mining and NLP. Attempts were made earlier to generate extractive text summary for Kannada and Telugu languages[6,7]. We aim to combine IE methods to summarization by using tagging rules like named entity recognition (NER). The objectives of the study are:

- To develop abstractive content-aware summary.
- To retrieve the content relevant to aspects of each category.
- To develop a method of creating different sentences.
- To generate simple, easy to understand, conveyable and cohesive text.

Here we describe the steps of abstractive summarizer such as preprocessing, categorization, attribute extraction and summary generation based on templates[8]. Figure 1 shows the flow diagram of the proposed system.

The input document could be a news article, product description, or biography of some eminent personality. The text document is pre-processed by lemmatization and stemming with Parts of Speech (POS) tagging using a cross lingual tool[9]. Identification of named entities such as names, locations, dates, etc. forms an integral part of this phase. Repositories of rules and gazetteers are complied to assist entity identification in NER phase.

Categorization acts as an indication of the context of information that is to be included in the summary. The document is classified to specific category based on the information to be extracted. TF/IDF rule based classifier is used for classification. After eliminating stop words and their variations, frequency and relevance of terms is determined. These frequent terms are compared with category-specific keywords to determine the category of the document.

Once the document is categorized, attribute extraction modules associated with that category are applied.

The methodology interprets predictable elements called attributes, which follow a guide called class. The class presents a very specific, unified information model of the given topic. Attributes are category-specific, primary pieces of information that are assumed to be present in the summary.

Classes are configurations that indicate multiple attributes to be identified for a given topic. The IE rules extract candidate replies for these attributes. Several classes can be merged to handle documents belonging to more than one category. TF/IDF classifier is used to categorize the document, which in turn determines the classes to be applied to it.

The POS and NE tags along with synonymous verb and noun forms help in crafting the IE rules. Gazetteers and

*For correspondence. (e-mail: jagadish.k@msrit.edu)

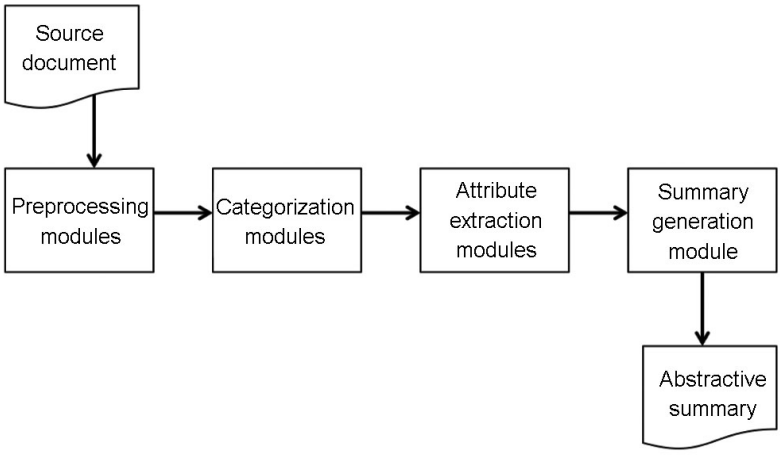

Figure 1. Overall system flow diagram.

NAME: Name of the person
PLACE: Place of Birth
DOB: Date of Birth
DOD: Date of Demise
AWARDS: Accolades and awards given

PEN NAME: Author's assumed name
WORKS: Literary works of the Author

Rule <- <name1> का/की कल्पित नाम <name2>
his/her assumed name
Author's name <- <name1>
Penname <- <name2>

Figure 2. Instance of a class for the category biography.

noun inflections are used to extract the information by including them in the IE rules.

Figure 2 gives a class for the category Biography. Apart from attributes applicable to all biographies, specialized class for literary personalities can be combined with the generic Biography class. The class has multiple IE rules to extract the attributes. In the example above, the rule attempts to identify the pen name of the author.

Redundant information may be identified by IE modules which serve to reinforce the validity and salience of the attribute chosen. Content selection heuristics are used and precise information piece for summary generation is chosen.

The key information expected in the summary for a given topic remains static and hence no reconstruction of attributes is required. Only the IE rules need to be inflexed to accommodate the variations of other languages.

Filling the template with the identified summary is the final stage. Templates are natural language generators that map their non-linguistic input to linguistic structure. Templates are generic structures of sentences with important pieces of information. The attributes extracted in the previous stage are mapped to deliver the information in an effective manner.

Template-based sentence generation requires the extracted information piece to be compounded with the right inflections. This triggers root word extraction and appropriate transformations on attributes to facilitate the completion of sentences in the template.

A probable drawback of using template-based sentence generation is monotony in the structure of the generated summaries. A comprehensive set of guided templates may help in generating variety of sentences and deliverables based on the category.

The proposed IE rule-based approach attempts to extract suitable information using lexical analysis tools like POS tagging and NER. This ensures an information rich summary that reduces redundancy in the information conveyed. The algorithm is as follows:

- Perform POS Tagging and Stemming on input text document.
- Recognize named entities like person, locations, dates, etc. using gazetteers and rules.
  - ♦ Identify category of the text document using statistical methods like TF.
  - ♦ Extract information for Aspects of the corresponding scheme using IE rules.
- Select appropriate template and populate it to generate a summary.

Sample template and the generated summary for Biography category in Telugu language is shown in Figure 3. The angular brackets indicate the position of different attributes to be replaced in the template. The underlined text in the summary indicates the attributes extracted from the original document.

The system depends on domain knowledge, shallow NLP and hand-written IE rules. It can be expanded to cover a plethora of focus groups. The system makes a clear distinction between the NLP stage and just extraction of keywords from a given text input. This allows

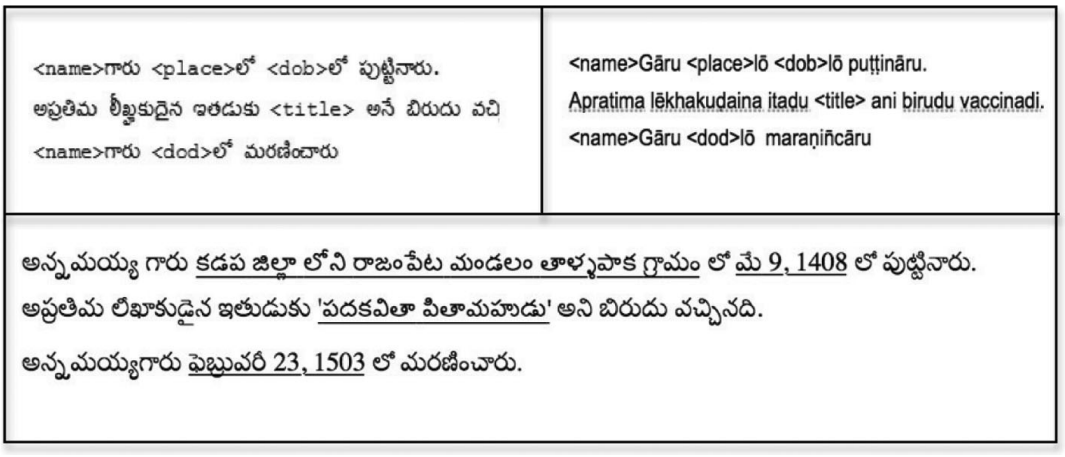

Figure 3. Sample Telugu template and its generated summary.

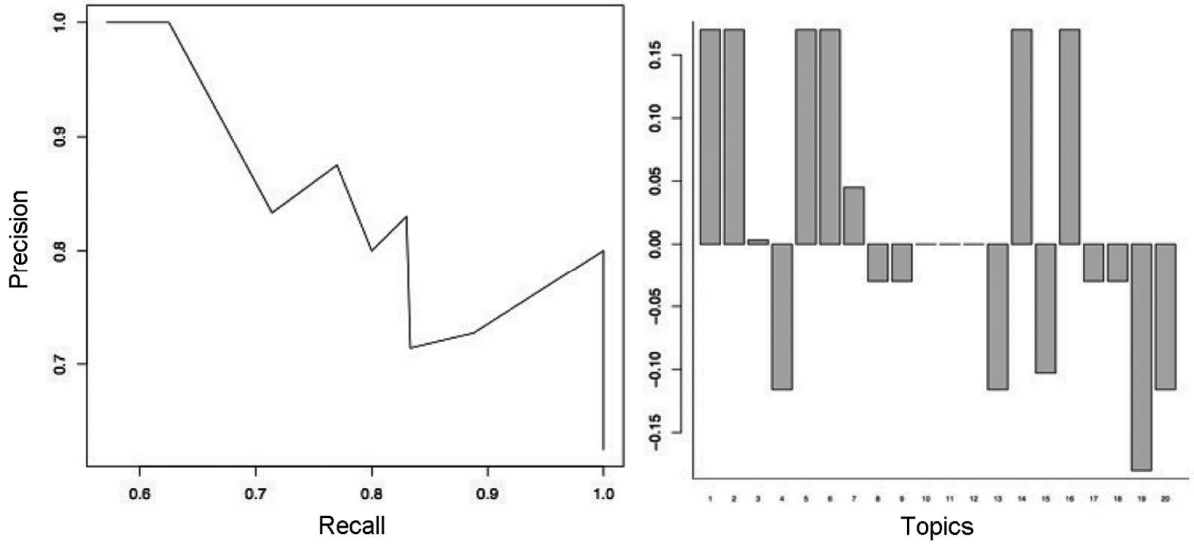

Figure 4. Recall–precision graph and average precision histogram.

Table 1. Evaluation results of the developed abstractive system

Category	Precision	Recall	Accuracy	*F*-measure
Biographies	0.9011	0.7313	0.772	0.81
Natural disasters	1.0001	0.9224	0.923	0.96
Reviews of products	1.0100	0.9125	0.911	0.95
Cultural events	0.7101	1.0216	0.714	0.83
Cricket	0.7930	0.6761	0.575	0.73
Attacks	0.7714	0.5202	0.435	0.61
Average	0.8642	0.7973	0.7217	0.815

integration of new information extraction rules and guided methods with existing ones. The current implementation designs are reusable to many topics.

A single standard rubric or metric to measure the performance of automatic text summarization is unavailable. Generally, manual evaluation is performed by humans who are informed to estimate the quality of a system, based on a number of criteria[10]. The criteria considered are precision, recall, accuracy and F-measure.

Table 1 shows the evaluation of the system over six different categories. Fifteen human judges were given the task of identifying attributes in each category. The number of attributes recognized for a category remains constant irrespective of length of the input document.

Figure 4 shows the recall-precision graph for 30 documents in different categories. The graph shows good precision and recall values. The average precision of a run on each topic against the median average precision of all corresponding runs on the topic is measured by the average precision histogram.

The evaluation of generated summaries is hard to understand due to non availability of a generic abstractive text summarization system for Indian languages. Hence, comparison with summaries generated by an extractive summarizer is made in extrinsic evaluation. In extraction

Table 2. Evaluation of extractive summarizer

Category	Precision	Recall	Accuracy	*F*-measure
Biography	0.27	0.6	0.44	0.37
Cricket	0.80	1.0	0.80	0.88
Natural disasters	0.38	1.0	0.46	0.55
Average	0.4833	0.867	0.567	0.6

Table 3. Abstractive versus extractive test summarization

Type	Precision	Recall	Accuracy	*F*-measure
Abstractive	0.8642	0.7973	0.7217	0.815
Extractive	0.4833	0.867	0.567	0.6

methods, determining text units by considering the lexical and statistical relevance or by matching phrase patterns[11] is emphasized. The extractive summarizer uses General Social Survey (GSS) coefficients and TF/IDF methods for extracting keywords to generate the summary.

The evaluation is conducted on three different categories by limiting the extract to the top ten highly ranked sentences. This has been compared with the abstractive summary which has ten sentences in it.

Table 2 shows the performance of extractive summarizer for the above four criteria.

A comparison between the evaluations of proposed abstractive summary with existing extractive summary is shown in Table 3. It is noticed that the extractive summarizer has high recall but compromises on precision. The proposed system performs better over the three categories considered when compared with extractive summary. It is also evident from Table 3 that in terms of crispness, information coverage, compression ratio and readability, abstractive is efficient over extractive summary.

This communication presents the methodology to create abstractive summaries of text documents written in Indian regional languages. The methodology has proved to have good precision values and readability. The text summarization problem is modelled as an IR problem. Distinction between IE and NLG stages allows the addition of new classes, IE rules, and improvements in each stage without affecting the other stages. The challenges in Indian languages are handled at each stage by writing IE rules and creating generic templates.

Template-based models generate flatness and monotony in the summary generated. To resolve this monotony, WordNet[12] (freely available lexical database) or Simple NLG[13] (Java API) may be suggested to facilitate the generation of natural language. The speech output of summaries can be explored as an extension of the presented work in future.

1. Kumar, M., Das, D. and Rudnicky, A. I., Summarizing non-textual events with a 'briefing' focus. In Proceedings of Recherche d'Information Assistée par Ordinateur, Pittsburgh, USA, 30 May–1 June 2007.
2. Jayashree, R., Srikanta Murthy, K. and Sunny, K., Keyword extraction based summarization of categorized Kannada text documents. *Int. J. Soft Comput.*, 2011, **2**(4).
3. Sarkar, K., Bengali text summarization by sentence extraction. In Proceedings of International Conference on Business and Information Management, NIT, Durgapur, 2012, pp. 233–245.
4. Embar, V. R., Deshpande, S. R., Vaishnavi, A. K., Jain, V., Kallimani, J. S., sArAmsha – a Kannada abstractive summarizer. In Proceedings of International Conference on Advances in Computing, Communications and Informatics, Mysore, 22–25 August 2013.
5. Das, A. and Bandyopadhyay, S., Syntactic sentence fusion techniques for Bengali. *Proc. Int. J. Computer Sci. Inf. Technol.*, 2011, **2**(1), 494–503.
6. Kallimani, J. S., Srinivasa, K. G., Eswara Reddy, B., Information retrieval by text summarization for an Indian regional language. In 6th International Conference on Natural Language Processing and Knowledge Engineering, Beijing, China, IEEE NLP-KE 2010, 21–23 August 2010, IEEE Catalog Number: CFP10811-PRT, ISBN:978-1-4244-6897-3, pp. 596–599.
7. Kallimani, J. S., Srinivasa, K. G. and Eswara Reddy, B., Information extraction by an abstractive text summarization for an Indian regional language. In 7th International Conference on Natural Language Processing and Knowledge Engineering, Tokushima, Japan, IEEE NLP-KE 2011, 27–29 November 2011.
8. Genest, P.-E. and Lapalme, G., Text generation for abstractive summarization. In Proceedings of the Third Text Analysis Conference, National Institute of Standards and Technology, Maryland, USA, 2010.
9. Reddy, S. and Sharoff, S., Cross language POS taggers (and other tools) for Indian languages: an experiment with Kannada using Telugu resources. In Proceedings of IJCNLP Workshop on Cross Lingual Information Access: Computational Linguistics and the Information Need of Multilingual Societies. Chiang Mai, Thailand, 2011.
10. John Dragomir R. Radev, Hovy, E. and McKeown, K., *Introduction to the Special Issue on Summarization*, Association for Computational Linguistics, 2002, vol. 28, no. 4; doi: http://dx.doi.org/10.1162/089120102762671927.
11. Bruce Hahn, U. and Mani, I., The challenges of automatic summarization. *IEEE-Comput.*, 2000, **33**(11), 29–36; doi: http://dx.doi.org/10.1109/2.881692.
12. George, A., Miller, WordNet: a lexical database for English. *Commun. ACM*, 1995, **38**(11), 39–41; doi: http://dx.doi.org/10.1145/219717.219748.
13. Gatt and Reiter, E., Simple NLG: a realization engine for practical applications. In Proceedings of ENLG, 2009.

Received 3 September 2014; accepted 30 August 2015

doi: 10.18520/cs/v110/i1/69-72