Multi-Document Summarization using Automatic Key-Phrase Extraction

Pinaki Bhaskar
Department of Computer Science & Engineering,
Jadavpur University, Kolkata – 700032, India
pinaki.bhaskar@gmail.com

Abstract

The development of a multi-document summarizer using automatic key-phrase extraction has been described. This summarizer has two main parts; first part is automatic extraction of Key-phrases from the documents and second part is automatic generation of a multi-document summary based on the extracted key-phrases. The CRF based Automatic Key-phrase extraction system has been used here. A document graph-based topic/query focused automatic multi-document summarizer is used for summarization where extracted key-phrases are used as topic. The summarizer has been tested on the standard TAC 2008 test data sets of the Update Summarization Track. Evaluation using the ROUGE-1.5.5 tool has resulted in ROUGE-2 and ROUGE–SU-4 scores of 0.10548 and 0.13582 respectively.

1 Introduction

Text Summarization, as the process of identifying the most salient information in a document or set of documents (for multi document summarization) and conveying it in less space, became an active field of research in both Information Retrieval (IR) and Natural Language Processing (NLP) communities. Summarization shares some basic techniques with indexing as both are concerned with identification of the essence of a document. Also, high quality summarization requires sophisticated NLP techniques in order to deal with various Parts Of Speech (POS) taxonomy and inherent subjectivity. Typically, one may distinguish various types of summarizers.

Multi document summarization requires creating a short summary from a set of documents, which concentrate on the same topic. Sometimes an additional query is also given to specify the information need of the summary. Generally, an effective summary should be relevant, concise and fluent. It means that the summary should cover the most important concepts in the original document set, contains less redundant information and should be well organized.

In this paper, we proposes a multi-document summarizer, based on key-phrase extraction, clustering technique and sentence fusion. Unlike traditional extraction based summarizers, which do not take into consideration the inherent structure of the document, our system will add structure to documents in the form of graph. During initial preprocessing, text fragments are identified from the documents, which constitute the nodes of the graph. Edges are defined as the correlation measure between nodes of the graph. We define our text fragments as sentence.

First, during preprocessing stage it performs some document-based tasks like identifying seed summary nodes and constructing graph over them. Then key-phrase extraction module extracts the key-phrases form the documents and it performs key-phrase search over the cluster to find a sentence identifying relevant phrases. With the relevant phrases, the new compressed sentence has been constructed and then fused for summary. The performance of the system depends much on the identification of relevant phrases and compression of the sentences where the previous one again highly depends on the key-phrase extraction module.

Although, we have presented all the examples in the current discussion for English language only, we argue that our system can be adapted to work on other language (i.e. Hindi, Bengali etc.) with some minor addition in the system like incorporating language dependent stop word list, the stemmer and the parser for the language.

Proceedings of the Student Research Workshop associated with RANLP 2013, pages 22–29,
Hissar, Bulgaria, 9-11 September 2013.

2 Related Work

Currently, most successful multi-document summarization systems follow the extractive summarization framework. These systems first rank all the sentences in the original document set and then select the most salient sentences to compose summaries for a good coverage of the concepts. For the purpose of creating more concise and fluent summaries, some intensive post-processing approaches are also appended on the extracted sentences. For example, redundancy removal (Carbonell and Goldstein, 1998) and sentence compression (Knight and Marcu, 2000) approaches are used to make the summary more concise. Sentence re-ordering approaches (Barzilay et al., 2002) are used to make the summary more fluent. In most systems, these approaches are treated as independent steps. A sequential process is usually adopted in their implementation, applying the various approaches one after another.

A lot of research work has been done in the domain of multi-document summarization (both query dependent and independent). MEAD (Radev et al., 2004) is a centroid based multi document summarizer, which generates summaries using cluster centroids produced by topic detection and tracking system. NeATS (Lin and Hovy, 2002) selects important content using sentence position, term frequency, topic signature and term clustering. XDoX (Hardy et al., 2002) identifies the most salient themes within the document set by passage clustering and then composes an extraction summary, which reflects these main themes.

Graph-based methods have been proposed for generating query independent summaries. Websumm (Mani and Bloedorn, 2000) uses a graph-connectivity model to identify salient information. Zhang et al. (2004) proposed the methodology of correlated summarization for multiple news articles. In the domain of single document summarization a system for query-specific document summarization has been proposed (Varadarajan and Hristidis, 2006) based on the concept of document graph. A document graph-based query focused multi-document summarization system is described by Bhaskar and Bandyopadhyay, (2010a and 2010b). In the present work, the same summarization approach has been followed. As this summarizer is query independent, it extract the key-phrases and then the extracted key-phrases are used as query or keywords.

Works on identification of key-phrase using noun phrase are reported in (Barker and Cornacchia, 2000). Noun phrases are extracted from a text using a base noun phrase skimmer and an off-the-shelf online dictionary. Key-phrase Extraction Algorithm (KEA) was proposed in order to automatically extract key-phrase (Witten et al., 1999). The supervised learning methodologies have also been reported (Frank et al, 1999). Some works have been done for automatic keywords extraction using CRF technique. A comparative study on the performance of the six keyword extraction models, i.e., CRF, SVM, MLR, Logit, BaseLine1 and BaseLine2 has been reported in (Chengzhi et al., 2008). The study shows that CRF based system outperforms SVM based system. Bhaskar and Bandyopadhyay (2012) have developed a supervised system for automatic extraction of Key-phrases using Conditional Random Fields (CRF).

First a key-phrase extraction system has been developed based on the Bhaskar and Bandyopadhyay's (2012) method. Then a graph-based summarization system has been developed, where the key-phrase extraction system has been integrated for extraction key-phrases from document, which are serves as query or topic during summary generation.

3 Document-Based Process

3.1 Graph-Based Clustered Model

The proposed graph-based multi-document summarization method consists of following steps:

(1) The document set $D = \{d_1, d_2, \ldots d_n\}$ is processed to extract text fragments, which are sentences in this system as it has been discussed earlier. Let for a document d_i, the sentences are $\{s_{i1}, s_{i2}, \ldots s_{im}\}$. Each text fragment becomes a node of the graph.

(2) Next, edges are created between nodes across the documents where edge score represents the degree of correlation between inter-documents nodes.

(3) Seed nodes are extracted which identify the relevant sentences within D and a search graph is built to reflect the semantic relationship between the nodes.

(4) At query time, each node is assigned a key-phrase dependent score and the search graph is expanded.

(5) A key-phrase dependent multi-document summary is generated from the search graph.