Language Independent Extractive Summarization

Rada Mihalcea
Department of Computer Science and Engineering
University of North Texas
rada@cs.unt.edu

Abstract

We demonstrate *TextRank* – a system for unsupervised extractive summarization that relies on the application of iterative graph-based ranking algorithms to graphs encoding the cohesive structure of a text. An important characteristic of the system is that it does not rely on any language-specific knowledge resources or any manually constructed training data, and thus it is highly portable to new languages or domains.

1 Introduction

Given the overwhelming amount of information available today, on the Web and elsewhere, techniques for efficient automatic text summarization are essential to improve the access to such information. Algorithms for extractive summarization are typically based on techniques for sentence extraction, and attempt to identify the set of sentences that are most important for the understanding of a given document. Some of the most successful approaches to extractive summarization consist of supervised algorithms that attempt to learn what makes a good summary by training on collections of summaries built for a relatively large number of training documents, e.g. (Hirao et al., 2002), (Teufel and Moens, 1997). However, the price paid for the high performance of such supervised algorithms is their inability to easily adapt to new languages or domains, as new training data are required for each new type of data. *TextRank* (Mihalcea and Tarau, 2004), (Mihalcea, 2004) is specifically designed to address this problem, by using an extractive summarization technique that does not require any training data or any language-specific knowledge sources. *TextRank* can be effectively applied to the summarization of documents in different languages without any modifications of the algorithm and without any requirements for additional data. Moreover, results from experiments performed on standard data sets have demonstrated that the performance of *TextRank* is competitive with that of some of the best summarization systems available today.

2 Extractive Summarization

Ranking algorithms, such as Kleinberg's $HITS$ algorithm (Kleinberg, 1999) or Google's $PageRank$ (Brin and Page, 1998) have been traditionally and successfully used in Web-link analysis, social networks, and more recently in text processing applications. In short, a graph-based ranking algorithm is a way of deciding on the importance of a vertex within a graph, by taking into account global information recursively computed from the entire graph, rather than relying only on local vertex-specific information. The basic idea implemented by the ranking model is that of *voting* or *recommendation*. When one vertex links to another one, it is basically casting a vote for that other vertex. The higher the number of votes that are cast for a vertex, the higher the importance of the vertex.

These graph ranking algorithms are based on a random walk model, where a walker takes random steps on the graph, with the walk being modeled as a Markov process – that is, the decision on what edge to follow is solely based on the vertex where the walker is currently located. Under certain conditions, this

Proceedings of the ACL Interactive Poster and Demonstration Sessions,
pages 49–52, Ann Arbor, June 2005.

model converges to a stationary distribution of probabilities associated with vertices in the graph, representing the probability of finding the walker at a certain vertex in the graph. Based on the Ergodic theorem for Markov chains (Grimmett and Stirzaker, 1989), the algorithms are guaranteed to converge if the graph is both aperiodic and irreducible. The first condition is achieved for any graph that is a non-bipartite graph, while the second condition holds for any strongly connected graph. Both these conditions are achieved in the graphs constructed for the extractive summarization application implemented in *TextRank*.

While there are several graph-based ranking algorithms previously proposed in the literature, we focus on two algorithms, namely $PageRank$ (Brin and Page, 1998) and $HITS$ (Kleinberg, 1999).

Let $G = (V, E)$ be a directed graph with the set of vertices V and set of edges E, where E is a subset of $V \times V$. For a given vertex V_i, let $In(V_i)$ be the set of vertices that point to it (predecessors), and let $Out(V_i)$ be the set of vertices that vertex V_i points to (successors).

2.1 PageRank

$PageRank$ (Brin and Page, 1998) is perhaps one of the most popular ranking algorithms, and was designed as a method for Web link analysis. Unlike other graph ranking algorithms, $PageRank$ integrates the impact of both incoming and outgoing links into one single model, and therefore it produces only one set of scores:

$$PR(V_i) = (1 - d) + d * \sum_{V_j \in In(V_i)} \frac{PR(V_j)}{|Out(V_j)|} \quad (1)$$

where d is a parameter that is set between 0 and 1, and has the role of integrating random jumps into the random walking model.

2.2 HITS

$HITS$ (Hyperlinked Induced Topic Search) (Kleinberg, 1999) is an iterative algorithm that was designed for ranking Web pages according to their degree of “authority”. The $HITS$ algorithm makes a distinction between “authorities” (pages with a large number of incoming links) and “hubs” (pages with a large number of outgoing links). For each vertex, $HITS$ produces two sets of scores – an “authority” score, and a “hub” score:

$$HITS_A(V_i) = \sum_{V_j \in In(V_i)} HITS_H(V_j) \quad (2)$$

$$HITS_H(V_i) = \sum_{V_j \in Out(V_i)} HITS_A(V_j) \quad (3)$$

Starting from arbitrary values assigned to each node in the graph, the ranking algorithm iterates until convergence below a given threshold is achieved. After running the algorithm, a score is associated with each vertex, which represents the *importance* of that vertex within the graph. Note that the final values are not affected by the choice of the initial value, only the number of iterations to convergence may be different.

When the graphs are built starting with natural language texts, it may be useful to integrate into the graph model the *strength* of the connection between two vertices V_i and V_j, indicated as a weight w_{ij} added to the corresponding edge. Consequently, the ranking algorithm is adapted to include edge weights, e.g. for $PageRank$ the score is determined using the following formula (a similar change can be applied to the $HITS$ algorithm):

$$PR^W(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} w_{ji} \frac{PR^W(V_j)}{\sum_{V_k \in Out(V_j)} w_{kj}} \quad (4)$$

While the final vertex scores (and therefore rankings) for weighted graphs differ significantly as compared to their unweighted alternatives, the number of iterations to convergence and the shape of the convergence curves is almost identical for weighted and unweighted graphs.

For the task of single-document extractive summarization, the goal is to rank the sentences in a given text with respect to their importance for the overall understanding of the text. A graph is therefore constructed by adding a vertex for each sentence in the text, and edges between vertices are established using sentence inter-connections, defined using a simple similarity relation measured as a function of content overlap. Such a relation between two sentences can be seen as a process of *recommendation*: a sentence that addresses certain concepts in a text gives the reader a *recommendation* to refer to other sentences in the text that address the same concepts, and therefore a link can be drawn between any two such sentences that share common content.

The overlap of two sentences can be determined simply as the number of common tokens between the lexical representations of the two sentences, or it can be run through filters that e.g. eliminate stopwords, count only words of a certain category, etc. Moreover, to avoid promoting long sentences, we use a normalization factor and divide the content overlap of two sentences with the length of each sentence.

The resulting graph is highly connected, with a weight associated with each edge, indicating the strength of the connections between various sentence pairs in the text. The graph can be represented as: (a) simple *undirected* graph; (b) directed weighted graph with the orientation of edges set from a sentence to sentences that follow in the text (*directed forward*); or (c) directed weighted graph with the orientation of edges set from a sentence to previous sentences in the text (*directed backward*).

After the ranking algorithm is run on the graph, sentences are sorted in reversed order of their score, and the top ranked sentences are selected for inclusion in the summary. Figure 1 shows an example of a weighted graph built for a short sample text.

[1] Watching the new movie, “Imagine: John Lennon,” was very painful for the late Beatle’s wife, Yoko Ono.
[2] “The only reason why I did watch it to the end is because I’m responsible for it, even though somebody else made it,” she said.
[3] Cassettes, film footage and other elements of the acclaimed movie were collected by Ono.
[4] She also took cassettes of interviews by Lennon, which were edited in such a way that he narrates the picture.
[5] Andrew Solt (“This Is Elvis”) directed, Solt and David L. Wolper produced and Solt and Sam Egan wrote it.
[6] “I think this is really the definitive documentary of John Lennon’s life,” Ono said in an interview.

3 Evaluation

English document summarization experiments are run using the summarization test collection provided in the framework of the Document Understanding Conference (DUC). In particular, we use the data set of 567 news articles made available during the DUC 2002 evaluations (DUC, 2002), and the corresponding 100-word summaries generated for each of these documents. This is the single document summarization task undertaken by other systems participating in the DUC 2002 document summarization evaluations.

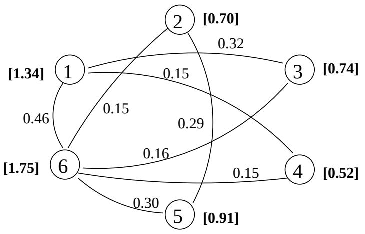

Figure 1: Graph of sentence similarities built on a sample text. Scores reflecting sentence importance are shown in brackets next to each sentence.

To test the language independence aspect of the algorithm, in addition to the English test collection, we also use a Brazilian Portuguese data set consisting of 100 news articles and their corresponding manually produced summaries. We use the TeMário test collection (Pardo and Rino, 2003), containing newspaper articles from online Brazilian newswire: 40 documents from *Jornal de Brasil* and 60 documents from *Folha de São Paulo*. The documents were selected to cover a variety of domains (e.g. world, politics, foreign affairs, editorials), and manual summaries were produced by an expert in Brazilian Portuguese. Unlike the summaries produced for the English DUC documents – which had a length requirement of approximately 100 words, the length of the summaries in the TeMário data set is constrained relative to the length of the corresponding documents, i.e. a summary has to account for about 25-30% of the original document. Consequently, the automatic summaries generated for the documents in this collection are not restricted to 100 words, as in the English experiments, but are required to have a length comparable to the corresponding manual summaries, to ensure a fair evaluation.

For evaluation, we are using the ROUGE evaluation toolkit[1], which is a method based on Ngram statistics, found to be highly correlated with human evaluations (Lin and Hovy, 2003). The evaluation is done using the Ngram(1,1) setting of ROUGE, which was found to have the highest correlation with human judgments, at a confidence level of 95%.

Table 2 shows the results obtained on these two data sets for different graph settings. The table also lists baseline results, obtained on summaries generated by

[1] ROUGE is available at http://www.isi.edu/˜cyl/ROUGE/.

Algorithm	Graph		
	Undirected	Forward	Backward
$HITS_A^W$	0.4912	0.4584	**0.5023**
$HITS_H^W$	0.4912	**0.5023**	0.4584
$PageRank^W$	0.4904	0.4202	**0.5008**
Baseline	0.4799		

Table 1: English single-document summarization.

Algorithm	Graph		
	Undirected	Forward	Backward
$HITS_A^W$	0.4814	0.4834	0.5002
$HITS_H^W$	0.4814	0.5002	0.4834
$PageRank^W$	0.4939	0.4574	**0.5121**
Baseline	0.4963		

Table 2: Portuguese single-document summarization.

taking the first sentences in each document. By ways of comparison, the best participating system in DUC 2002 was a *supervised* system that led to a ROUGE score of 0.5011.

For both data sets, *TextRank* applied on a *directed backward* graph structure exceeds the performance achieved through a simple (but powerful) baseline. These results prove that graph-based ranking algorithms, previously found successful in Web link analysis and social networks, can be turned into a state-of-the-art tool for extractive summarization when applied to graphs extracted from texts. Moreover, due to its unsupervised nature, the algorithm was also shown to be language independent, leading to similar results and similar improvements over baseline techniques when applied on documents in different languages. More extensive experimental results with the *TextRank* system are reported in (Mihalcea and Tarau, 2004), (Mihalcea, 2004).

4 Conclusion

Intuitively, iterative graph-based ranking algorithms work well on the task of extractive summarization because they do not only rely on the local context of a text unit (vertex), but they also take into account information recursively drawn from the entire text (graph). Through the graphs it builds on texts, a graph-based ranking algorithm identifies connections between various entities in a text, and implements the concept of *recommendation*. In the process of identifying important sentences in a text, a sentence recommends other sentences that address similar concepts as being useful for the overall understanding of the text. Sentences that are highly recommended by other sentences are likely to be more informative for the given text, and will be therefore given a higher score.

An important aspect of the graph-based extractive summarization method is that it does not require deep linguistic knowledge, nor domain or language specific annotated corpora, which makes it highly portable to other domains, genres, or languages.

Acknowledgments

We are grateful to Lucia Helena Machado Rino for making available the TeMário summarization test collection and for her help with this data set.

References

S. Brin and L. Page. 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7).

DUC. 2002. Document understanding conference 2002. http://www-nlpir.nist.gov/projects/duc/.

G. Grimmett and D. Stirzaker. 1989. *Probability and Random Processes*. Oxford University Press.

T. Hirao, Y. Sasaki, H. Isozaki, and E. Maeda. 2002. Ntt's text summarization system for duc-2002. In *Proceedings of the Document Understanding Conference 2002*.

J.M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632.

C.Y. Lin and E.H. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of Human Language Technology Conference (HLT-NAACL 2003)*, Edmonton, Canada, May.

R. Mihalcea and P. Tarau. 2004. TextRank – bringing order into texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, Barcelona, Spain.

R. Mihalcea. 2004. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Lingusitics (ACL 2004) (companion volume)*, Barcelona, Spain.

T.A.S. Pardo and L.H.M. Rino. 2003. TeMario: a corpus for automatic text summarization. Technical report, NILC-TR-03-09.

S. Teufel and M. Moens. 1997. Sentence extraction as a classification task. In *ACL/EACL workshop on "Intelligent and scalable Text summarization"*, pages 58–65, Madrid, Spain.