FastSum:
Fast and accurate query-based multi-document summarization

Frank Schilder and **Ravikumar Kondadadi**
Research & Development
Thomson Corp.
610 Opperman Drive, Eagan, MN 55123, USA
FirstName.LastName@Thomson.com

Abstract

We present a fast query-based multi-document summarizer called FastSum based solely on word-frequency features of clusters, documents and topics. Summary sentences are ranked by a regression SVM. The summarizer does not use any expensive NLP techniques such as parsing, tagging of names or even part of speech information. Still, the achieved accuracy is comparable to the best systems presented in recent academic competitions (i.e., Document Understanding Conference (DUC)). Because of a detailed feature analysis using Least Angle Regression (LARS), FastSum can rely on a minimal set of features leading to fast processing times: *1250* news documents in *60* seconds.

1 Introduction

In this paper, we propose a simple method for effectively generating query-based multi-document summaries without any complex processing steps. It only involves sentence splitting, filtering candidate sentences and computing the word frequencies in the documents of a cluster, topic description and the topic title. We use a machine learning technique called regression SVM, as proposed by (Li et al., 2007). For the feature selection we use a new model selection technique called Least Angle Regression (LARS) (Efron et al., 2004).

Even though machine learning approaches dominated the field of summarization systems in recent DUC competitions, not much effort has been spent in finding simple but effective features. Exceptions are the SumBasic system that achieves reasonable results with only one feature (i.e., word frequency in document clusters) (Nenkova and Vanderwende, 2005). Our approach goes beyond SumBasic by proposing an even more powerful feature that proves to be the best predictor in all three recent DUC corpora. In order to prove that our feature is more predictive than other features we provide a rigorous feature analysis by employing LARS.

Scalability is normally not considered when different summarization systems are compared. Processing time of more than several seconds per summary should be considered unacceptable, in particular, if you bear in mind that using such a system should help a user to process lots of data faster. Our focus is on selecting the minimal set of features that are computationally less expensive than other features (i.e., full parse). Since FastSum can rely on a minimal set of features determined by LARS, it can process *1250* news documents in *60* seconds.[1] A comparison test with the MEAD system[2] showed that FastSum is more than 4 times faster.

2 System description

We use a machine learning approach to rank all sentences in the topic cluster for summarizability. We use some features from Microsoft's PYTHY system (Toutonova et al., 2007), but added two new features, which turned out to be better predictors.

First, the pre-processing module carries out tokenization and sentence splitting. We also created a sentence simplification component which is based

[1] 4-way/2.0GHz PIII Xeon 4096Mb Memory

[2] http://www.summarization.com/mead/

Proceedings of ACL-08: HLT, Short Papers (Companion Volume), pages 205–208,
Columbus, Ohio, USA, June 2008. ©2008 Association for Computational Linguistics

on a few regular expressions to remove unimportant components of a sentence (e.g., *As a matter of fact,*). This processing step does not involve any syntactic parsing though. For further processing, we ignore all sentences that do not have at least two exact word matches or at least three fuzzy matches with the topic description.[3]

Features are mainly based on word frequencies of words in the clusters, documents and topics. A cluster contains 25 documents and is associated with a topic. The topic contains a topic title and the topic descriptions. The topic title is list of key words or phrases describing the topic. The topic description contains the actual query or queries (e.g., *Describe steps taken and worldwide reaction prior to introduction of the Euro on January 1, 1999.*).

The features we used can be divided into two sets; word-based and sentence-based. Word-based features are computed based on the probability of words for the different containers (i.e., cluster, document, topic title and description). At runtime, the different probabilities of all words in a candidate sentence are added up and normalized by length. Sentence-based features include the length and position of the sentence in the document. The starred features **1** and **4** are introduced by us, whereas the others can be found in earlier literature.[4]

- ***1** *Topic title frequency* (1): ratio of number of words t_i in the sentence s that also appear in the topic title $\mathcal{T}$ to the total number of words $t_{1..|s|}$ in the sentence s: $\frac{\sum_{i=1}^{|s|} f_{\mathcal{T}}(t_i)}{|s|}$, where $f_{\mathcal{T}} = \begin{cases} 1 & : \ t_i \in \mathcal{T} \\ 0 & : \ otherwise \end{cases}$

- **2** *Topic description frequency* (2): ratio of number of words t_i in the sentence s that also appear in the topic description $\mathcal{D}$ to the total number of words $t_{1..|s|}$ in the sentence s: $\frac{\sum_{i=1}^{|s|} f_{\mathcal{D}}(t_i)}{|s|}$, where $f_{\mathcal{D}} = \begin{cases} 1 & : \ t_i \in \mathcal{D} \\ 0 & : \ otherwise \end{cases}$

- **3** *Content word frequency*(3): the average content word probability $p_c(t_i)$ of all content words $t_{1..|s|}$ in a sentence s. The content word probability is defined as $p_c(t_i) = \frac{n}{N}$, where n is the number of times the word occurred in the cluster and N is the total number of words in the cluster: $\frac{\sum_{i=1}^{|s|} p_c(t_i)}{|s|}$

- ***4** *Document frequency* (4): the average document probability $p_d(t_i)$ of all content words $t_{1..|s|}$ in a sentence s. The document probability is defined as $p_d(t_i) = \frac{d}{D}$, where d is the number of *documents* the word t_i occurred in for a given cluster and D is the total number of documents in the cluster: $\frac{\sum_{i=1}^{|s|} p_d(t_i)}{|s|}$

The remaining features are *Headline frequency* (**5**), *Sentence length* (**6**), *Sentence position (binary)* (**7**), and *Sentence position (real)* (**8**)

Eventually, each sentence is associated with a score which is a linear combination of the above mentioned feature values. We ignore all sentences that do not have at least two exact word matches.[5] In order to learn the feature weights, we trained a SVM on the previous year's data using the same feature set. We used a regression SVM. In regression, the task is to estimate the functional dependence of a dependent variable on a set of independent variables. In our case, the goal is to estimate the score of a sentence based on the given feature set. In order to get training data, we computed the word overlap between the sentences from the document clusters and the sentences in DUC model summaries. We associated the word overlap score to the corresponding sentence to generate the regression data. As a last step, we use the pivoted QR decomposition to handle redundancy. The basic idea is to avoid redundancy by changing the relative importance of the rest of the sentences based on the currently selected sentence. The final summary is created from the ranked sentence list after the redundancy removal step.

3 Results

We compared our system with the top performing systems in the last two DUC competitions. With our best performing features, we get ROUGE-2 (Lin, 2004) scores of 0.11 and 0.0925 on 2007 and 2006

[3]Fuzzy matches are defined by the OVERLAP similarity (Bollegala et al., 2007) of at least 0.1.

[4]The numbers are used in the feature analysis, as in figure 2.

[5]This threshold was derived experimentally with previous data.

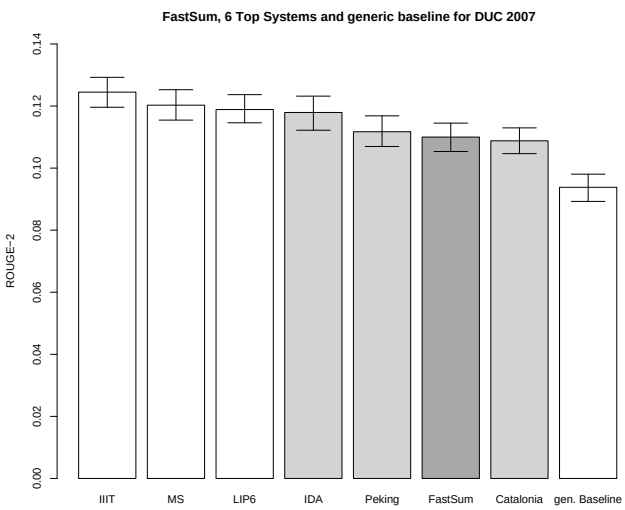

Figure 1: ROUGE-2 results including 95%-confidence intervals for the top 6 systems, FastSum and the generic baseline for DUC 2007

DUC data, respectively. These scores correspond to rank 6th for DUC 2007 and the 2nd rank for DUC 2006. Figure 1 shows a graphical comparison of our system with the top 6 systems in DUC 2007. According to an ANOVA test carried out by the DUC organizers, these 6 systems are significant better than the remaining 26 participating systems.

Note that our system is better than the PYTHY system for 2006, if no sentence simplification was carried out (DUC 2006: 0.089 (without simplification); 0.096 (with simplification)). Sentence simplification is a computationally expensive process, because it requires a syntactic parse.

We evaluated the performance of the FastSum algorithm using each of the features separately. Table 1 shows the ROUGE score (recall) of the summaries generated when we used each of the features by themselves on 2006 and 2007 DUC data, trained on the data from the respective previous year. Using only the Document frequency feature by itself leads to the second best system for DUC 2006 and to the tenth best system for DUC 2007.

This first simple analysis of features indicates that a more rigorous feature analysis would have benefits for building simpler models. In addition, feature selection could be guided by the complexity of the features preferring those features that are computationally inexpensive.

Feature name	2007	2006
Title word frequency	0.096	0.0771
Topic word frequency	0.0996	0.0883
Content word frequency	0.1046	0.0839
Document frequency	**0.1061**	**0.0903**
Headline frequency	0.0938	0.0737
Sentence length	0.054	0.0438
Sentence position(binary)	0.0522	0.0484
Sentence position (real-valued)	0.0544	0.0458

Table 1: ROUGE-2 scores of individual features

We chose a so-called model selection algorithm to find a minimal set of features. This problem can be formulated as a shrinkage and selection method for linear regression. The Least Angle Regression (LARS) (Efron et al., 2004) algorithm can be used for computing the least absolute shrinkage and selection operator (LASSO) (Tibshirani, 1996).At each stage in LARS, the feature that is most correlated with the response is added to the model. The coefficient of the feature is set in the direction of the sign of the feature's correlation with the response.

We computed LARS on the DUC data sets from the last three years. The graphical results for 2007 are shown in figure 2. In a LARS graph, features are plotted on the x-axis and the corresponding coefficients are shown on y-axis. The value on the x-axis is the ratio of norm of the coefficent vector to the maximal norm with no constraint. The earlier a feature appears on the x-axis, the better it is. Table 2 summarizes the best four features we determined with LARS for the three available DUC data sets.

Year	Top Features
2005	4 2 5 1
2006	4 3 2 1
2007	4 3 5 2

Table 2: The 4 top features for the DUC 2005, 2006 and 2007 data

Table 2 shows that feature 4, document frequency, is consistently the most important feature for all three data sets. Content word frequency (3), on the other hand, comes in as second best feature for 2006 and 2007, but not for 2005. For the 2005 data, the Topic description frequency is the second best feature. This observation is reflected by our single fea-

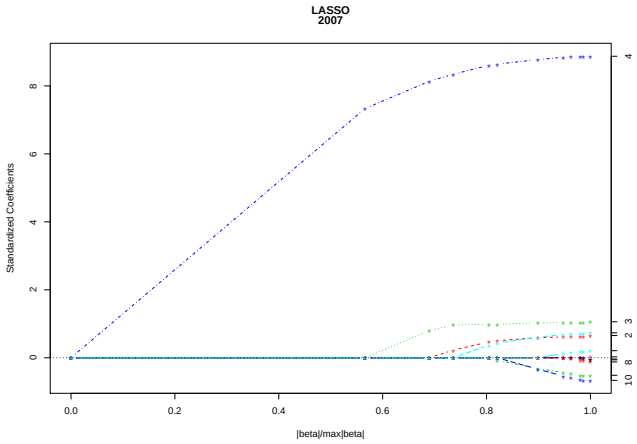

Figure 2: Graphical output of LARS analysis:
Top features for 2007: 4 Document frequency, 3 Content word frequency, 5 Headline frequency, 2 Topic description frequency

ture analysis for DUC 2006, as shown in table 1. Similarly, Vanderwende et al. (2006) report that they gave the Topic description frequency a much higher weight than the Content word frequency.

Consequently, we have shown that our new feature Document frequency is consistently the best feature for all three past DUC corpora.

4 Conclusions

We proposed a fast query-based multi-document summarizer called FastSum that produces state-of-the-art summaries using a small set of predictors, two of those are proposed by us: document frequency and topic title frequency. A feature analysis using least angle regression (LARS) indicated that the document frequency feature is the most useful feature consistently for the last three DUC data sets. Using document frequency alone can produce competitive results for DUC 2006 and DUC 2007. The two most useful feature that takes the topic description (i.e., the queries) into account is based on the number of words in the topic description and the topic title. Using a limited feature set of the 5 best features generates summaries that are comparable to the top systems of the DUC 2006 and 2007 main task and can be generated in real-time, since no computationally expensive features (e.g., parsing) are used.

From these findings, we draw the following conclusions. Since a feature set mainly based on word frequencies can produce state-of-the-art summaries, we need to analyze further the current set-up for the query-based multi-document summarization task. In particular, we need to ask the question whether the selection of relevant documents for the DUC topics is in any way biased. For DUC, the document clusters for a topic containing relevant documents were always pre-selected by the assessors in preparation for DUC. Our analysis suggests that simple word frequency computations of these clusters and the documents alone can produce reasonable summaries. However, the human selecting the relevant documents may have already influenced the way summaries can automatically be generated. Our system and systems such as SumBasic or SumFocus may just exploit the fact that relevant articles prescreened by humans contain a high density of good content words for summarization.[6]

References

D. Bollegala, Y. Matsuo, and M. Ishizuka. 2007. Measuring Semantic Similarity between Words Using Web Search Engines. In *Proc. of 16th International World Wide Web Conference (WWW 2007)*, pages 757–766, Banff, Canada.

B. Efron, T. Hastie, I.M. Johnstone, and R. Tibshirani. 2004. Least angle regression. *Annals of Statistics*, 32(2):407–499.

S. Gupta, A. Nenkova, and D. Jurafsky. 2007. Measuring Importance and Query Relevance in Topic-focused Multi-document Summarization. In *Proc. of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 193–196, Prague, Czech Republic.

S. Li, Y. Ouyang, W. Wang, and B. Sun. 2007. Multi-document summarization using support vector regression. In *Proceedings of DUC 2007, Rochester, USA*.

C. Lin. 2004. Rouge: a package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*.

A. Nenkova and L. Vanderwende. 2005. The impact of frequency on summarization. In *MSR-TR-2005-101*.

R. Tibshirani. 1996. Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc B.*, 58(1):267–288.

K. Toutonova, C. Brockett, J. Jagarlamudi, H. Suzuko, and L. Vanderwende. 2007. The PYTHY Summarization System: Microsoft Research at DUC2007. In *Proc. of DUC 2007*, Rochester, USA.

L. Vanderwende, H. Suzuki, and C. Brockett. 2006. Microsoft Research at DUC 2006: Task-focused summarization with sentence simplification and lexical expansion. In *Proc. of DUC 2006, New York, USA*.

[6]Cf. Gupta et al. (2007) who come to a similar conclusion by comparing between word frequency and log-likelihood ratio.