Learning to Summarise XML Documents using Content and Structure

Massih R. Amini[†]   Anastasios Tombros[*]   Nicolas Usunier[†]   Mounia Lalmas[*]   Patrick Gallinari[†]
amini@poleia.lip6.fr   *tassos@dcs.qmul.ac.uk*   *usunier@poleia.lip6.fr*   *mounia@dcs.qmul.ac.uk*   *gallinari@poleia.lip6.fr*

[†]University Pierre and Marie Curie
8, rue du capitaine Scott
75015, Paris
France

[*]Queen Mary, University of London
Mile End Road
London E1 4NS
United Kingdom

ABSTRACT

Documents formatted in eXtensible Markup Language (XML) are becoming increasingly available in collections of various document types. In this paper, we present an approach for the summarisation of XML documents. The novelty of this approach lies in that it is based on features not only from the content of documents, but also from their logical structure. We follow a machine learning like, sentence extraction-based summarisation technique. To find which features are more effective for producing summaries this approach views sentence extraction as an ordering task. We evaluated our summarisation model using the INEX dataset. The results demonstrate that the inclusion of features from the logical structure of documents increases the effectiveness of the summariser, and that the learnable system is also effective and well-suited to the task of summarisation in the context of XML documents.

Categories and Subject Descriptors

H.4.m [**Information Systems**]: [Miscellaneous-Text Summarisation]; I.2.6 [**Computing Methodologies**]: [Learning-Parameter]

Keywords

Summarisation, machine learning, ranking algorithms, XML documents, content and structure features

1. INTRODUCTION

With the growing availability of on-line text resources, it has become necessary to provide users with systems that obtain answers to queries in a manner which is both efficient and effective. Single document text summarisation (SDS) can be coupled with conventional search engines and help users to evaluate the relevance of documents [5] for providing answers to their queries.

In this paper we follow a summarisation approach that is based on the machine learning (ML) paradigm and investigate the effectiveness of an XML summarisation approach by combining structural and content features to extract sentences for summaries. Like most previous work on ML for summarisation we rely on supervised learning, where a set of training documents and their extract summaries are available. We explore an ML approach for SDS based on the Area under the ROC curve (AUC). The main rationale of this approach is to automatically combine different features,

each being a numerical representation of a given extraction criterion. The summariser learns how to best combine sentence features based on its effectiveness at assigning higher scores to summary sentences than to non-summary ones.

The contributions of this work are therefore twofold: first, we propose and justify the effectiveness of a new algorithm that optimises the AUC ordering criterion, instead of the mostly used classification error criterion in ML approaches for SDS [1], and second, we investigate the summarisation of XML documents by taking into account features relating both to the content and the logical structure of the documents.

2. TRAINABLE TEXT SUMMARIZERS

In order to evaluate which of the classification or the AUC criteria are better suited to the SDS task, we used in our experiments the same logistic model in both frameworks.

Logistic model for classification
In the particular case of a logistic classifier, one makes the following assumption on the form of the posterior probability of the class *relevant* given a sentence $s = (s_1, ..., s_n)$ represented by a vector of scores: $p(relevant \mid s) = \frac{1}{1+e^{-2\sum_{i=1}^{n} \lambda_i s_i}}$. The parameters of the feature combination $\Lambda = (\lambda_1, ..., \lambda_n)$ can then be learnt by maximizing the binomial log-likelihood:

$$\mathcal{L}(\mathcal{D}; \Lambda) = -\frac{1}{|\mathcal{S}|} \sum_{y \in \{-1,1\}} \sum_{\mathbf{s} \in \mathcal{S}^y} \log(1 + e^{-2y \sum_{i=1}^{n} \lambda_i s_i}) \quad (1)$$

where $\mathcal{D}$ is the set of training documents, $\mathcal{S}^1$ and $\mathcal{S}^{-1}$ are respectively the set of relevant and irrelevant sentences in the training set and $|\mathcal{S}|$ is the number of sentences in the aforesaid set.

Logistic model for AUC
The logistic assumption, adapted to AUC, becomes $p(relevant | \mathbf{s}, \mathbf{s}') = \frac{1}{1+e^{-2\sum_{i=1}^{n} \theta_i (s_i - s'_i)}}$. The binomial log-likelihood for AUC is given by:

$$\mathcal{L}_{\mathcal{A}}(\mathcal{D}; \Theta) = -\frac{1}{|\mathcal{S}^{-1}||\mathcal{S}^{1}|} \sum_{(\mathbf{s}, \mathbf{s}') \in \mathcal{S}} \log(1 + e^{-2(H(s) - H(s'))}) \quad (2)$$

Following [3], one can asymptotically show that the population minimizers of the expected binomial log-likelihood for AUC and $E\left[e^{H(s') - H(s)}\right]$ coincide. This is an interesting finding which reinforces the duality between classification and AUC.

3. SUMMARISING XML DOCUMENTS

In this paper, we take the logical structure of documents into account when producing summaries, as well as the content, and we

learn an effective combination of features for summarisation. The structural features we use in our approach are (i) the depth of the element in which the sentence is contained (e.g. section, subsection, subsubsection, etc.), (ii) the sibling number of the element in which the sentence is contained (e.g. 1st, middle, last), (iii) the number of sibling elements of the element in which the sentence is contained and (iv) the position in the element of the paragraph in which the sentence is contained (e.g. first, or not).

Our basic content-only query (COQ) comprises terms in the title of the document (*Title* query), as well as the title keywords augmented by the most frequent terms in the document (up to 10 such terms) (*Title-MFT* query). The importance of title terms for SDS can also be extended to components of finer granularity (e.g. sections, subsections, etc.), by using the title of the document to find relevant sentences within any component, or, where appropriate, by using meaningful titles of components.

Since the *Title* query may be very short, we have also employed query-expansion techniques such as Local Context Analysis (LCA) or thesaurus expansion methods (i.e. WordNet), we also included two queries using word clusters. This is another source of information about the relevance of sentences to summaries. It is a more contextual approach compared to the title-based queries, as it seeks to take advantage of the co-occurrence of terms within sentences all over the corpus, as opposed to the local information provided by the title-based queries. We used the cosine measure in order to compute a preliminary score between any sentence of a document and these generic queries. The scoring measure doubles the cosine scoring of sentences containing acronyms or cue-terms.

4. EXPERIMENTS AND RESULTS

In our experiments, we used the INEX test collection and ran 3 algorithms - a logistic model optimising the ordering AUC metric using an iterative scaling scheme, a logistic classifier optimising the classification binomial log-likelihood criteria (1) and the Rank-Boost algorithm [2]. To measure the effect of structure features we have learnt the best learning algorithm using COQ features alone and COQ features with the aforementioned structure features.

To obtain sentence-based extract summaries for all articles in both datasets, for training and evaluation purposes, we applied an algorithm proposed by Marcu [4] in order to generate extracts from the abstracts. This algorithm has shown a high degree of correlation to sentence extracts produced by humans. We therefore evaluate the effectiveness of our learning algorithm on the basis of how well it matches the automatic extracts.

In figure 1 we present the precision and recall graph that we obtained through the combination of content and structure features for the INEX dataset when using the three learning algorithms.

A first, non-surprising, result is that the combination of features by learning outperforms each feature alone. The results also show that the two ordering algorithms are more effective than the logistic classifier. However, the comparison between the AUC algorithm and the logistic classifier leads to the conclusion that an ordering criterion is better suited to SDS than a classification criterion.

When comparing the two ordering algorithms, we see that the AUC algorithm slightly outperforms the RankBoost algorithm for high recall values. Since both ordering algorithms optimise the same criteria, the difference in performance can be explained by the class of functions that each algorithm learns. The RankBoost algorithm outputs a nonlinear combination of the features while with the AUC algorithm we obtain a linear combination of these features. As the space of features is small, the non-linear Rank-Boost model has low bias and high variance and hence attempts to overfit the data.
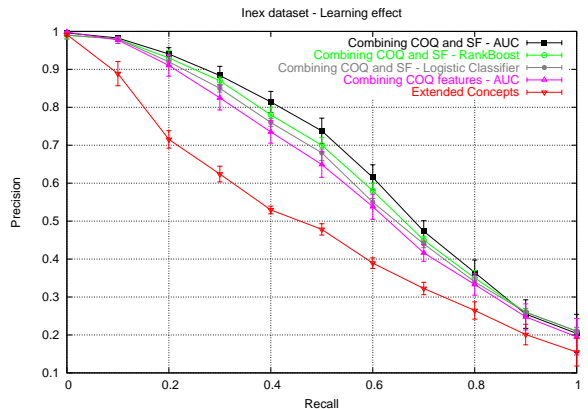

Figure 1: Precision-Recall curves at 10% compression ratio for the learning effects on INEX dataset. Each point represents the mean performance for 10 cross-validation folds. The bars show standard deviations for the estimated performance.

5. CONCLUSION

The results that we presented in the previous section are encouraging in relation to our two main motivations: a novel learning algorithm for SDS, and the inclusion of structure, in addition to content, features for the summarisation of XML documents. The ultimate aim of our approach for the summarisation of XML documents is to produce summaries for components at any level of granularity (e.g. section, subsection, etc.). The content and structure features that we used can be applied to any level of granularity. In particular, the most effective content (expanded concepts with word clusters and projected concepts on word clusters), and structure features (depth of element and position of paragraph in the element), can be applied to various granularity levels within an XML tree. By looking at the results of this study as a whole, we can say that the work presented here achieved its main aim, to effectively summarise XML documents by combining content and structure features through using novel machine learning approaches. This work has however a greater impact, as we believe that it can be applied to datasets containing documents of other types. The availability of XML data will continue to increase as, for example, XML is becoming the W3C standard for representing documents (e.g. in digital libraries where content can be of any type). The availability of intelligent summarisation approaches for XML data with therefore become increasingly important, and we believe that this work has provided a step towards this direction.

6. REFERENCES

[1] Amini M.-R., Patrick Gallinari: The use of unlabeled data to improve supervised learning for text summarization, ACM SIGIR, 105–112, (2002).

[2] Freund Y., Iyer R., Schapire R.E., Singer Y. An efficient boosting algorithm for combining preferences, JMLR, Vol. 4, 933–969, (2003).

[3] Friedman J., Hastie T., Tibshirani R.: Additive Logistic Regression: a Statistical View of Boosting, TR, (1998).

[4] Marcu D.: The Automatic Construction of Large-Scale Corpora for Summarization Research, ACM SIGIR, 137-144, (1999).

[5] Tombros A., Sanderson M.: Advantages of query biased summaries in information retrieval, ACM SIGIR, 2–10, (1999).