Combinations of Different Raters for Text Summarization

Róbert Móro[*]
Institute of Informatics and Software Engineering
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Ilkovičova 3, 842 16 Bratislava, Slovakia
xmoror@stuba.sk

Abstract

Because of the unprecedented growth of information on the web it became difficult to decide what information is truly relevant. We present a method of personalized summarization based on a combination of different raters that process the additional information, such as user characteristics or document metadata to extract from the document information that is important for a particular user or in a chosen domain, thus decreasing the information overload. We have experimented with the proposed method in the adaptive web-based educational system ALEF. Our results suggest that user-added annotations can be used to improve the summaries' quality compared to the generic variant.

Categories and Subject Descriptors

H.3.3 [**Information and Storage Retrieval**]: Information Search and Retrieval—*information filtering, selection process*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*text analysis*; K.3.3 [**Computers and Education**]: Computer Uses in Education—*computer assisted instruction (CAI)*

Keywords

automatic text summarization, personalization, annotations, relevant domain terms, web-based learning

1. Introduction

As we can find almost everything on the web, it has become very problematic to find what we actually want or need – to find relevant information. Moreover, fragments of relevant information are often scattered throught the entire document. We can considerably decrease the information overload by extracting these fragments (sentences) into a concise summary. This can be achieved automatically by a means of text summarization.

However, the term "relevant information" is subjective, because as users of the web, we differ in our interests, goals or knowledge. That is why we focus on the *personalized* text summarization. Using the additional information about users' characteristic we adapt the summaries to actual user's needs. With the arrival of Web 2.0, users are no longer passive consumers of Web content, but they can create new content and enrich the existing one by adding metadata, such as annotations (highlights, tags etc.). We use user-added annotations as another important source of personalization.

In the paper, we present a method of personalized text summarization that combines different sources of personalization. Our approach is domain-independent and it could be used in many domains, e.g. news portals or digital libraries. However, to evaluate the proposed method we have focused on the domain of learning, carrying out the experiments presented in this paper in the adaptive web-based educational system ALEF [10].

2. Related Work

A lot of research has been done in the field of automatic text summarization in the past fifty years. Gong and Liu [4] were first to use latent semantic analysis (LSA) for the text summarization. It finds salient topics or concepts in a document and includes into a summary sentences that capture the concepts (topics) of the document the best. Steinberger and Ježek [7] pointed out, that this approach fails to include sentences, which capture many concepts well, but have the highest score for none of them. In their modified approach sentences are selected based on their overall score computed as a combination of scores for each concept (topic). LSA has been used in other works as well, e.g. in [8], achieving comparable or better results than other approaches.

In order to improve summaries generated by the generic methods mentioned so far, we have to either employ more advanced methods of natural language processing or take into consideration other potentially useful information. In the former the goal is to produce *abstracts* not only extracts of sentences using paraphrasing, sentences compression or synonyms substitution. In the latter we can obtain much information in the form of *implicit feedback*.

[*] Master degree study programme in field Software Engineering. Supervisor: Professor Mária Bieliková, Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies, STU in Bratislava. Work described in this paper was presented at the 8th Student Research Conference in Informatics and Information Technologies IIT.SRC 2012.

Móro, R. Combinations of Different Raters for Text Summarization. Information Sciences and Technologies Bulletin of the ACM Slovakia, Special Section on Student Research in Informatics and Information Technologies, Vol. 4, No. 2 (2012) 56-58

Sun et al. [8] utilized clickthrough data which records how users find information through queries. If a user clicks on a link to a web page which is one of the results of her query, we can assume that terms from the query describe the page. We get even more accurate assumption if we consider other implicit indacators such as time spent on the web page, scrolling, copying [1], or even a user's gaze [5].

There have been some works done in the area of personalized summarization as well. Díaz et al. [2] personalized summarization to mirror users' long-term and short-term interests represented by a user model in the form of a vector of weighted keywords; disadvantage of this approach is that users had to manually insert keywords and weights into the model. However, what types of information can be utilized to adapt the summaries and secondly, how to combine different sources of personalization remain open problems.

3. Method of Personalized Text Summarization

We present a method of personalized text summarization based on a method of latent semantic analysis [4], [7]. We have chosen LSA as a basis for our approach, because of its ability to provide better results compared to the other summarization methods. However, the proposed method of personalization could be used together with other generic summarization method as well.

The method consists of the following steps (based on LSA): The first one is pre-processing during which the terms are extracted from the document and the document's text is segmented to sentences. Then we construct a terms-sentences matrix which represents an input to singular value decomposition [4]:

$$\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathbf{T}} \tag{1}$$

where $\mathbf{A}$ is an input terms-sentences matrix, $\mathbf{U}$ is a matrix the columns of which represent left singular vectors and rows terms of the document, while the columns of matrix $\mathbf{V}$ represent right singular vectors and its rows sentences of the document. $\mathbf{\Sigma}$ is a diagonal matrix with diagonal elements being non-negative singular values in descending order representing relative relevance of the identified concepts in the document. The final step is a selection of sentences; we select sentences with the highest score using approach proposed by Steinberger and Ježek in [9].

We have identified a construction of a terms-sentences matrix as a step suitable for summary personalization. In this step terms extracted from the document are assigned their respective weights. We construct *a personalized terms-sentences matrix* using our proposed weighting scheme which extends the conventional weighting scheme based on tf-idf method by linear combination of multiple raters:

$$w(t_{ij}) = \sum_k \alpha_k R_k(t_{ij}) \tag{2}$$

where $w(t_{ij})$ is a weight of a term t_{ij} in the matrix and α_k is a linear coefficient of a rater R_k.

We have designed a set of *generic* and *personalized* raters which positively or negatively affect the weight of each term. The design of *terms frequency rater* and *terms location rater* has been inspired by Luhn [6] and Edmundson [3] respectively and we use them to produce baseline generic variants of summarization.

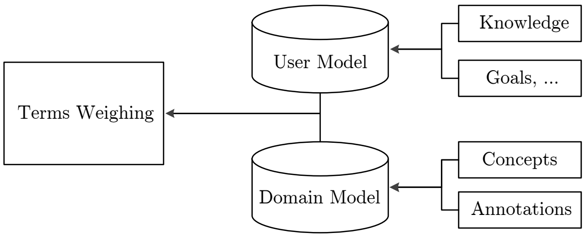


Figure 1: Inputs of terms weighing in the construction of a personalized matrix.

In order to adapt summaries, we can use information from user and domain model (see Figure 1). Considering the domain of learning, we have identified three main sources of personalization and adaptation:

- *Domain conceptualization* – domain models capture experts' knowledge of the domain in the form of important concepts and relationships among them; instead of heavyweight ontologies we can use relevant domain terms as a lightweight semantic to represent concepts [9].
- *Knowledge of the users* – it is recorded in overlay user models of adaptive educational systems; we can use it for example to extract from a document information that a user has already learned, but needs to revise it before an exam or a test, i.e. it is suitable for knowledge revision scenario.
- *User-added annotations* – when a user highlights a fragment of text, we assume that the fragment contains information deemed important or interesting by the user; when many users highlight the same (or similar) fragment of text, we assume that the fragment contains important and valuable information in general.

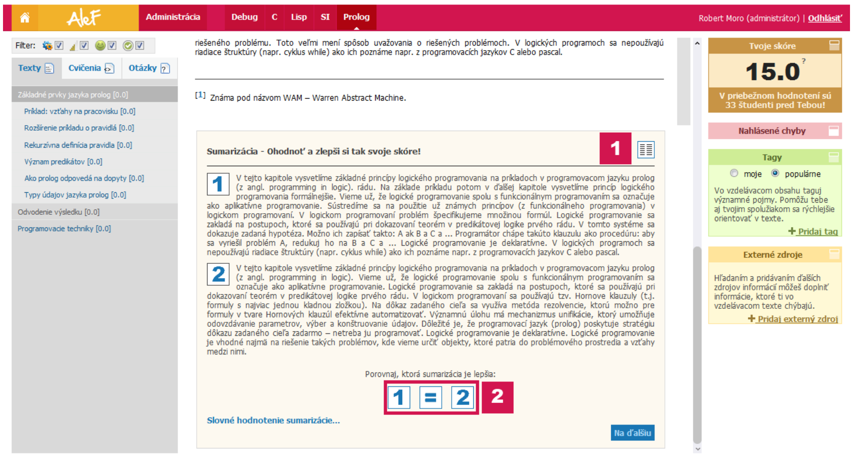
We have designed the raters processing the identified sources of personalization and adaptation, i.e. *relevant domain terms rater*, *knowledge rater* and *annotations rater*. They determine which terms are important based on their source of information and assign them higher weights.

4. Evaluation

We have implemented the presented method as a REST (*Representational State Transfer*) web service. Clients can communicate with the summarizer using a JSON protocol to set up a configuration of their own and request a summarization using the chosen configuration. It is domain and system-independent and it can be easily and transparently integrated with any system.

In order to evaluate the presented approach, we have integrated the implemented summarizer with the adaptive web-based educational system ALEF [10] by means of a summarization widget (see Figure 2). We have experimentally evaluated the summarization using user-added annotations in comparison with the generic variant.

In total, around 20 students of the *Functional and Logic Programming* course have taken part in this experiment. Their task has been to evaluate the presented summary on a five-point scale. Moreover, we have chosen an expert group of five students who have been presented both


AleF
Administrácia
Debug
C
Lisp
SI
Prolog
Robert Moro (administrátor) | Odhlásiť
Filter: ☑ ☑ ☑ ☑
Texty
Cvičenia
Otázky
Základné prvky jazyka prolog [0.0]
Príklad: vzťahy na pracovisku [0.0]
Rozšírenie príkladu o pravidlá [0.0]
Rekurzívna definícia pravidla [0.0]
Význam predikátov [0.0]
Ako prolog odpovedá na dopyty [0.0]
Typy údajov jazyka prolog [0.0]
Odvodenie výsledku [0.0]
Programovacie techniky [0.0]
1
2
1
=
2
Slovné hodnotenie sumarizácie...
Na ďalšiu
Tvoje skóre
15.0
?
Nahlásené chyby
Tagy
○ moje
● populárne
+ Pridaj tag
Externé zdroje
+ Pridaj externý zdroj

riešeného problému. Toto veľmi mení spôsob uvažovania o riešených problémoch. V logických programoch sa nepoužívajú riadiace štruktúry (napr. cyklus while) ako ich poznáme napr. z programovacích jazykov C alebo pascal.

[1] Známa pod názvom WAM – Warren Abstract Machine.

Sumarizácia - Ohodnoť a zlepši si tak svoje skóre!

1 V tejto kapitole vysvetlíme základné princípy logického programovania na príkladoch v programovacom jazyku prolog (z angl. programming in logic). rádu. Na základe príkladu potom v ďalšej kapitole vysvetlíme princíp logického programovania formálnejšie. Vieme už, že logické programovanie spolu s funkcionálnym programovaním sa označuje ako aplikatívne programovanie. Sústredíme sa na použitie už známych princípov (z funkcionálneho programovania) v logickom programovaní. V logickom programovaní problém špecifikujeme množinou formúl. Logické programovanie sa zakladá na postupoch, ktoré sa používajú pri dokazovaní teorém v predikátovej logike prvého rádu. V tomto systéme sa dokazuje zadaná hypotéza. Možno ich zapísať takto: A ak B a C a ... Programátor chápe takúto klauzulu ako procedúru: aby sa vyriešil problém A, redukuj ho na B a C a ... Logické programovanie je deklaratívne. V logických programoch sa nepoužívajú riadiace štruktúry (napr. cyklus while) ako ich poznáme napr. z programovacích jazykov C alebo pascal.

2 V tejto kapitole vysvetlíme základné princípy logického programovania na príkladoch v programovacom jazyku prolog (z angl. programming in logic). Vieme už, že logické programovanie spolu s funkcionálnym programovaním sa označuje ako aplikatívne programovanie. Logické programovanie sa zakladá na postupoch, ktoré sa používajú pri dokazovaní teorém v predikátovej logike prvého rádu. V logickom programovaní sa používajú tzv. Hornove klauzuly (t.j. formuly s najviac jednou kladnou zložkou). Na dôkaz zadaného cieľa sa využíva metóda rezolvencie, ktorú možno pre formuly v tvare Hornových klauzúl efektívne automatizovať. Významnú úlohu má mechanizmus unifikácie, ktorý umožňuje odovzdávanie parametrov, výber a konštruovanie údajov. Dôležité je, že programovací jazyk (prolog) poskytuje stratégiu dôkazu zadaného cieľa zadarmo – netreba ju programovať. Logické programovanie je deklaratívne. Logické programovanie je vhodné najmä na riešenie takých problémov, kde vieme určiť objekty, ktoré patria do problémového prostredia a vzťahy medzi nimi.

Porovnaj, ktorá sumarizácia je lepšia:

V priebežnom hodnotení sú 33 študenti pred Tebou!

Vo vzdelávacom obsahu taguj významné pojmy. Pomôžu tebe aj tvojim spolužiakom sa rýchlejšie orientovať v texte.

Hľadaním a pridávaním ďalších zdrojov informácií môžeš doplniť informácie, ktoré ti vo vzdelávacom texte chýbajú.

Figure 2: Summarizer widget in the system ALEF: expert users evaluate summary variants by their comparison; they can switch between one and two-column view (1) and choose which variant is better or whether they are equal (2).

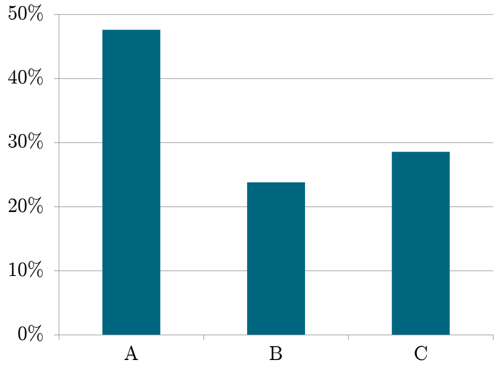


Figure 3: Comparison of summary variants: *A* means that summary considering annotations has been evaluated as better; *B* that generic variant has been evaluated as better and *C* that they has been evaluated as equal.

variants (summarization considering annotations as well as generic variant) in random order to decide which variant is better or whether they are equal.

The results of our experiment suggest that considering the annotations in the summarization process leads to better summaries compared to the generic variant (see Figure 3). Students – experts have evaluated the variant considering the annotations as better in 48% of the cases opposed to only 24% when they have evaluated as worse. The relatively higher percentage of cases when the variants have been evaluated as equal (29%) has been probably caused by the fact that not every document has been annotated (highlighted) by the users. The summarization considering annotations has also gained higher average score than the generic variant based on the five-point user rating.

5. Conclusions

Our proposed method of personalized summarization based on a combination of different raters allows us to consider various parameters or context of the summarization. The contribution lies also in the design of the specific raters, mainly the relevant domain terms rater and the annotations rater which we have also experimentally evaluated (the former not presented in the paper).

Possible future work is in automatic setting of the combination parameters. We can interpret these paramters as a measure of confidence in the specific raters and determine them dynamically based on the reliability of their sources of information or the category of the summarized document. For this purpose, different machine learning methods, such as decision trees or Bayesian networks might be employed.

Acknowledgements. This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/0675/11.

References

[1] M. Barla. Towards social-based user modeling and personalization. *Information Sciences and Technologies Bulletin of the ACM Slovakia*, 3(1):52–60, 2011.

[2] A. Díaz and P. Gervás. User-model based personalized summarization. *Information Processing and Management*, 43(6):1715–1734, 2007.

[3] H. P. Edmundson. New methods in automatic extracting. *J. of the Association for Computing Machinery*, 16(2):264–285, 1969.

[4] Y. Gong and X. Liu. Generic text summarization using relevance measure and latent semantic analysis. In *Proc. of the* 24^{th} *Int. ACM SIGIR Conf. on Research and Development in Inf. Retrieval*, pp. 19–25. Springer, 2001.

[5] M. Labaj. Automated acquisition of domain model for adaptive collaborative web-based learning. *Information Sciences and Technologies Bulletin of the ACM Slovakia*, 3(1):76–78, 2011.

[6] H. P. Luhn. The automatic creation of literature abstracts. *IBM J. of Research Development*, 2(2):159–165, 1958.

[7] J. Steinberger and K. Ježek. Text summarization and singular value decomposition. In *ADVIS '04: Proc. of Advances in Information Systems*, pp. 245–254. Springer, 2005.

[8] J.-T. Sun et al. Web-page summarization using clickthrough data. In *Proc. of the* 28^{th} *Int. ACM SIGIR Conf. on Research and Development in Inf. Retrieval*, pp. 194–201. ACM Press, 2005.

[9] M. Šimko. Automated acquisition of domain model for adaptive collaborative web-based learning. *Information Sciences and Technologies Bulletin of the ACM Slovakia*, 4(2), 2012.

[10] M. Šimko, M. Barla, and M. Bieliková. ALEF: A framework for adaptive web-based learning 2.0. In *Proc. of KCKS 2010, IFIP Advances in Information and Communication Technology*, pp. 367–378. Springer, 2010.