On the Evaluation of Snippet Selection for WebCLEF

Arnold Overwijk, Dong Nguyen, Claudia Hauff, Dolf Trieschnigg, Djoerd Hiemstra, and Franciska de Jong

University of Twente,
The Netherlands
arnold.overwijk@gmail.com, dong.p.ng@gmail.com,
c.hauff@ewi.utwente.nl, trieschn@ewi.utwente.nl,
hiemstra@cs.utwente.nl, f.m.g.dejong@ewi.utwente.nl

Abstract. WebCLEF is about supporting a user who is an expert in writing a survey article on a specific topic with a clear goal and audience by generating a ranked list with relevant snippets. This paper focuses on the evaluation methodology of WebCLEF. We show that the evaluation method and test set used for WebCLEF 2007 cannot be used to evaluate new systems and give recommendations how to improve the evaluation.

Keywords: Measurement, Performance, Experimentation.

1 Introduction

WebCLEF is about supporting a user who is writing an article and therefore wants to know more about a certain topic (i.e. undirected information search), which is the most common search goal [1]. This support consists of a list with relevant snippets. The degree to which the user's information need is satisfied is measured by the number of distinct atomic facts that the user includes in the article after analyzing the top snippets returned by the system.

The evaluation method should give insight into the parameters of the system and the performance of both participating and non-participating systems. In this paper we investigate the usefulness of the evaluation method of WebCLEF 2007 [2].

First, a brief overview of WebCLEF 2007's evaluation method is given, followed by a description of the experimental setup and the results. Based on the results, we propose a number of alternative evaluation methods. We finish with conclusions and possible future work.

2 Evaluation Method of WebCLEF 2007

The evaluation of WebCLEF relies on manual assessments created by the participants, who have manually selected the most relevant snippets from snippets delivered by the participating systems. The measures currently employed in the WebCLEF evaluation are *recall* and *precision*. Here, *recall* is defined as the sum of character lengths of all spans in the response of the system linked to nuggets (i.e. an aspect the

C. Peters et al. (Eds.): CLEF 2008, LNCS 5706, pp. 794–797, 2009.

user includes in his article), divided by the total sum of span lengths in the responses for a topic in all submitted runs. *Precision* is defined as the number of characters that belong to at least one span linked to a nugget, divided by the total character length of the system's response. More details about these measures as well as the data provided by WebCLEF can be found in the overview paper [2].

3 Experimental Setup

We investigate the evaluation method by creating several experimental systems. The general idea of our experiment is that if we can reason that a system is worse, almost equal or better than another system, this should also be reflected in the performance indicated by the evaluation method. As a baseline we use last year's best performing system, S_{base} [3]. We create three experimental systems that we argue to perform *worse*, *very similar* and *better* than this baseline, named S_{worse}, $S_{similar}$ and S_{better} respectively.

S_{worse} performs no sophisticated snippet selection. It simply delivers the snippets (i.e. paragraphs as in S_{base}) in order of occurrence; the first snippet is the first paragraph of the first document, etc. Therefore this system does much less than S_{base}, which orders snippets by relevance, removes redundant snippets, etc.

$S_{similar}$ gives almost identical output as S_{base}: it removes the last word of every snippet in the output of S_{base}. The amount of information returned to the user is almost the same when a snippet lacks only the last word, since the average length of a snippet is over 40 words. Obviously, $S_{similar}$ is not a realistic system but since it almost returns the same output as S_{base}, we argue that the evaluation metrics should return similar performance scores.

Initial experiments showed that S_{base}, performing best last year, actually contained a small programming error: only half of the intended stop word list was removed during a preprocessing step. Since it is not certain that the removal of the error leads to a better performing system, we compared S_{base} to two other systems, a system that filters all stop words and one that does not filter any stop words at all. One of these systems should perform better, whether filtering stop words is a good approach or not.

4 Results and Discussion

The measured performance of the evaluated systems are given in table 1.

Table 1. Performance of the experimental systems compared to the baseline

System	Precision	Recall	Rank
S_{base}	0.2018	0.2561	1
S_{worse}	0.0536	0.0680	5
$S_{similar}$	0.0597	0.0758	4
$S_{better\ –\ filtering\ stop\ words}$	0.1328	0.1685	2
$S_{better\ –\ not\ filtering\ stopwords}$	0.1087	0.1380	3

It is notable that the metric indicates that all systems perform worse than the baseline. Only S_{worse} meets our expectations; however, a more in dept analysis of the results tells us that simply returning the snippets in order of their occurrence results in the same performance as the baseline for six (i.e. topic 17, 18, 21, 23, 25, 26) out of thirty topics (20%). Moreover, the metric shows only a small performance difference between S_{worse} and $S_{similar}$. These results indicate that the available relevance judgments in combination with the evaluation methodology cannot be used to evaluate new systems.

An important problem of the evaluation metric is its strictness. According to the evaluation script a snippet from the manual assessments should exactly occur in the output of the system, otherwise there is no match at all. This explains why $S_{similar}$ has much lower performance scores. A slight change to the output of a perfect system results in a strong decrease of the measured performance.

Additionally, the pool of snippets to create relevance judgments was not very large, since there were only three participating systems. There might be snippets that are relevant to the user, but which are not delivered by one of the participants, resulting in incomplete relevance judgments. Such a setup gives a disadvantage to non-participating systems, since they might deliver such a snippet. This in combination with the strictness of the evaluation explains why S_{better} has lower performance scores. Notice that according to the evaluation metric, filtering only half of the intended stop word list performs better than filtering all stop words as well as not filtering any stop words at all. Again, the evaluation metric does not reflect the quality of the systems in its scores.

Furthermore, we noticed that some of the relevance judgments were not carefully created, which might influence the evaluation of new systems. For example some topics only contain non-relevant snippets (e.g. topic 14) and other topics do not contain any snippets at all (e.g. topic 12), which automatically results in a precision and recall value of zero. In topic 14 for example the user wants to find out if there are any blog search engines in Europe that are not subsidiaries of the big three search engines (Google, Yahoo! and Microsoft). Here the assessments file contains snippets like "blog search engines are hardly usable so far", which is not relevant to the user at all. This in combination with the strictness problem explains why the evaluation metric indicates that S_{worse} performs almost the same as $S_{similar}$. To be more precise, S_{base} provided for six topics exactly the same output as S_{worse}. Due to an error in the ranking algorithm no ranking could be determined for some topics and snippets were delivered in order of occurrence.

The pool problem can be solved with a larger number of participants. The problem with the manual assessments can also be solved with some effort, namely with multiple assessors per topic, which is already done in some other tracks (e.g. [4]). Unfortunately the strictness problem is not as easily solved, since the same information can be represented in several ways. The TREC QA task also has to deal with this problem [4]. However there are some existing evaluation methods that are less strict by calculating the amount of overlap.

One of them that is close to the current one, and therefore a reasonable solution, is already used in XML Retrieval [5]. In this approach the systems provide the offsets (i.e. the start and end of a passage in the document) of the delivered snippets from which the amount of overlap can be calculated to get an indication of the performance.

Another more common, approach for evaluating extractive summaries, which is the case in WebCLEF, is automatic comparison between reference and system summaries using n-grams. Originally this approach was applied to machine translation, but it has been developed in the ROUGE program for summary evaluation as well [6].

5 Conclusion and Future Work

For developers it is important to measure the system performance, especially in a task where it is hard to measure the quality of the output (i.e. WebCLEF). We explored several weaknesses in the evaluation method and the dataset of WebCLEF 2007. Unfortunately the evaluation does not provide information that is of the developers' interest nor does it reflect the performance of the system in a correct way. We showed that the manual assessments were not carefully created, which is mainly caused by the fact that it most of the times is very hard to judge whether a snippet is relevant to the user. Moreover we have shown that the measurement in general is not appropriate. With the current evaluation method a snippet in the assessments must occur exactly in the system's output. This is not realistic, since the same information can be variably expressed. A possible solution to this problem can be found in using n-grams (e.g. ROUGE [6]), because it is likely that the same information makes use of the same words. In addition it might be even better to combine this approach with TF.IDF measures to give different values to different n-grams. With such an approach words that occur less frequent, which are probably more specific and therefore contain more information, are given a higher value. We leave this question for future work.

Acknowledgments. This paper is based on research partly funded by IST project MESH (http://www.mesh-ip.eu) and by bsik program MultimediaN (http://www.multimedian.nl).

References

1. Rose, D.E., Levinson, D.: Understanding user goals in web search. In: Proceedings of the 13th international conference on World Wide Web. ACM, New York (2004)
2. Jijkoun, V., de Rijke, M.: Overview of WebCLEF 2007. In: Peters, C., Jijkoun, V., Mandl, T., Müller, H., Oard, D.W., Peñas, A., Petras, V., Santos, D. (eds.) CLEF 2007. LNCS, vol. 5152, pp. 725–731. Springer, Heidelberg (2008)
3. Jijkoun, V., de Rijke, M.: The University of Amsterdam at Web CLEF 2007: Using Centrality to Rank Web Snippets. In: CLEF 2007, Budapest, Hungary (2007)
4. Voorhees, E.M., Tice, D.M.: The TREC-8 question answering track evaluation. In: Text Retrieval Conference TREC-8, pp. 83–105 (1999)
5. Pehcevski, J., Thom, J.A.: HiXEval: Highlighting XML Retrieval Evaluation. In: Fuhr, N., Lalmas, M., Malik, S., Kazai, G. (eds.) INEX 2005. LNCS, vol. 3977, pp. 43–57. Springer, Heidelberg (2006)
6. Lin, C.-Y.: ROUGE: a Package for Automatic Evaluation of Summaries. In: Proceedings of Workshop on Text Summarization, Barcelona, Spain (2004)