KBAAA: A Web-based Toolkit for the Assessment and Analysis of Knowledge Base Acceleration Systems

Krisztian Balog
University of Stavanger
krisztian.balog@uis.no

Heri Ramampiaro
NTNU Trondheim
heri.ramampiaro@idi.ntnu.no

Kjetil Nørvåg
NTNU Trondheim
kjetil.norvag@idi.ntnu.no

ABSTRACT

In this paper we present KBAAA, a prototype system that provides a web-based interface for the analysis and assessment of knowledge base acceleration systems. KBAAA displays items from a voluminous document stream that are deemed central to a given target entity. Results can be visualised on a timeline along with the number of pageviews and edits made for that entity in the knowledge base. If available, the system also shows relevance assessments and computes evaluation metrics. Further functionality includes side-by-side comparison of two different approaches and free text search over the document collections. The prototype has been fully implemented and deployed on the dataset and target entities of the TREC 2012 KBA track.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information filtering

Keywords

Knowledge base acceleration, cumulative citation recommendation, information filtering

1. INTRODUCTION

Knowledge base acceleration (KBA) systems seek to help humans maintain and expand knowledge bases like Wikipedia by automatically recommending edits based on incoming content streams. In 2012, the Text REtrieval Conference (TREC) has launched a new Knowledge Base Acceleration track (TREC KBA[1]) with the aim to develop an experimental platform and evaluation methodology for the assessment of KBA systems. The first edition of the track focused on a single task, termed *cumulative citation recommendation* (CCR): filter a time-ordered corpus for documents that are highly relevant to a predefined set of target entities [4]. A new stream corpus was constructed specifically for this task that spans over a period of 7 months and contains over 400M documents.[2] Target entities (that serve as the input to the KBA system) are from Wikipedia; these are chosen such that they receive a moderate number of mentions in the stream corpus and have a complex network of relationships with other active entities. Given the sheer volume of the corpus, it not always easy to understand what is going on with a given entity during a particular time interval. There are a number of efforts towards building applications that allow for time-based exploration of news collections in an entity-oriented manner [1, 2, 5]. The KBAAA system we present here is an attempt towards building an intuitive interface for analysing document filtering techniques for KBA. Although KBAAA was developed specifically with the CCR task in mind, our goal is to make its major components be reusable for other tasks. Our prototype is available at http://research.idi.ntnu.no/wislab/kbaaa.

[1] http://trec-kba.org
[2] http://trec-kba.org/kba-stream-corpus-2012.shtml

OAIR'13, May 22-24, 2013, Lisbon, Portugal.

2. SYSTEM OVERVIEW

Figure 1 shows an overview of the KBAAA system architecture. The KBAAA system consists of two main parts: (i) a back-end that extracts and stores all information from the document collection, from KBA systems, and from the knowledge base, and (ii) a web-based front-end that serves as the user interface.

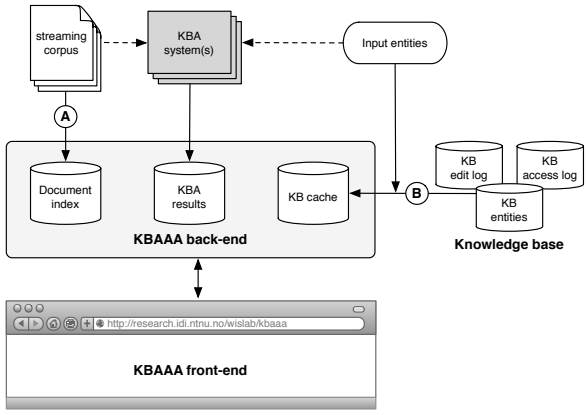

Figure 1: KBAAA system overview.

There are two main offline processing steps, indicated by the circled letters in Figure 1. (A) concerns the indexing of documents in the collection; (B) extracts information about the input entities from the knowledge base: the article describing the entity as well as time-stamped pageview counts and edit history (both of these are needed for the timeline visualisation). We present further details about the implementation of these steps in §4. It is important to emphasise that KBAAA does not perform the actual document

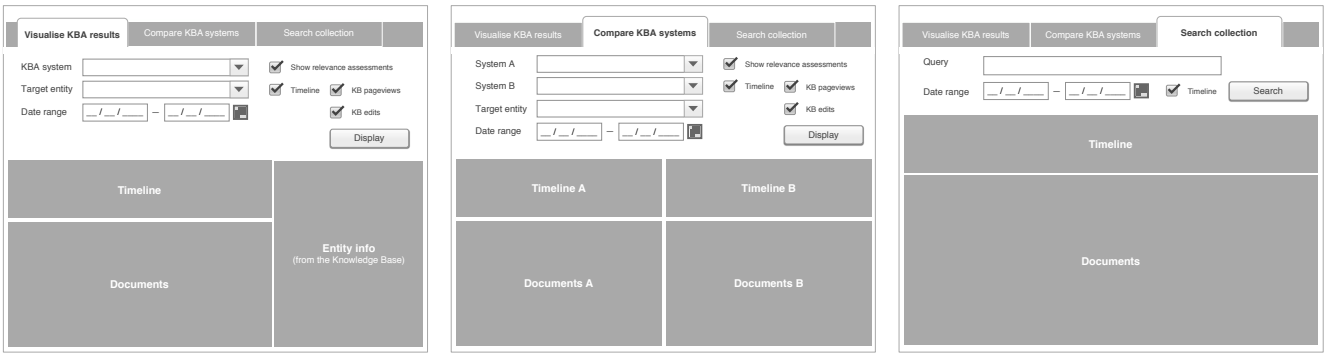

Figure 2: Schematic overview of KBAAA's functionality. (Left): visualising KBA results. (Middle): comparing KBA systems. (Right): searching the collection.

filtering for KBA (i.e., the CCR task); it merely provides an interface for analysing results generated by some external KBA system (these follow the runfile format of the TREC 2012 KBA track and have to be uploaded through the web interface). The next section introduces the functionality provided by the web-based front-end.

3. FUNCTIONALITY

KBAAA offers three main areas of functionality: visualising KBA results for a single system (§3.1), comparing two KBA systems (§3.2), and searching the collection (§3.3); see Figure 2.

3.1 Visualising KBA results

We show results for a given target entity, for a single KBA system. At its core, this is a ranked list of documents from a selected time period. To incorporate the temporal dimension, documents can be displayed on a timeline. The user can choose to have the pageviews and actual edits also appear on the same timeline; this helps to gain further insights into how a specific entity has evolved over a specific time period. In addition, information contained about the entity in the knowledge base is displayed on the right side. If available and selected, relevance assessments are shown for the individual documents; documents with annotator disagreements are marked in a special way, as these are cases that may require extra attention.

3.2 Comparing KBA systems

This view provides a side-by-side comparison of two KBA systems for the same target entity and time interval. It is created with the goal to help users analyse and understand differences between two systems and, as such, it offers several convenience features; for example, mouseover on a document immediately shows the ranks and confidence scores for that document for both systems as well as the corresponding relevance assessments. We can also compute and display various evaluation metrics for the two systems.

3.3 Searching the collection

We provide free text search functionality over the document collection, where results can be displayed on a timeline. Note that this view does not offer any specific features related to entities, but it has support for complex query operators.

4. IMPLEMENTATION

We now provide some details on the implementation of our system. The indexing and searching of documents is powered by Apache Solr. All other data, including (i) document metadata (URL, timestamp, and title), (ii) Wikipedia articles, page view statistics, and edit history for the input entities, (iii) CCR runs, and (iv) relevance assessments are stored in a MySQL database. All data-intensive preprocessing was done on Hadoop using MapReduce. The web-based interface is written in PHP and JavaScripts and runs under an Apache web server on a Linux machine.

5. CONCLUSIONS AND FUTURE WORK

We presented KBAAA, a working prototype developed for the assessment and analysis of knowledge base acceleration systems. The current version of our system uses the TREC 2012 KBA data collection and target entities. For demonstration purposes a number of KBA approaches we developed in [3] are made available. Additionally, registered users can upload results from their own KBA system and may choose to make these publicly available. It is our plan to provide support for future editions of the TREC KBA track.

References

[1] O. Alonso, K. Berberich, S. Bedathur, and G. Weikum. Time-based exploration of news archives. In *4th Workshop on Human-Computer Interaction and Information Retrieval (HCIR '10)*, 2010.

[2] K. Balog, M. de Rijke, R. Franz, H. Peetz, B. Brinkman, I. Johgi, and M. Hirschel. SaHaRa: Discovering entity-topic associations in online news. In *8th International Semantic Web Conference (ISWC '09)*, 2009.

[3] K. Balog, N. Takhirov, H. Ramampiaro, and K. Nørvåg. Multi-step classification approaches to cumulative citation recommendation. In *Open research Areas in Information Retrieval (OAIR '13)*, 2013.

[4] J. R. Frank, M. Kleiman-Weiner, D. A. Roberts, F. Niu, C. Zhang, C. Ré, and I. Soboroff. Building an entity-centric stream filtering test collection for TREC 2012. In *21th Text REtrieval Conference (TREC '12)*, 2013.

[5] M. Matthews, P. Tolchinsky, R. Blanco, J. Atserias, P. Mika, and H. Zaragoza. Searching through time in the New York Times. In *4th Workshop on Human-Computer Interaction and Information Retrieval (HCIR '10)*, 2010.