Entity-Based Semantic Search on Conversational Transcripts Semantic

Search on Hansard

Obinna Onyimadu[1,2], Keiichi Nakata[1], Ying Wang[1,2],
Tony Wilson[2], and Kecheng Liu[1]

[1] Informatics Research Centre, Henley Business School, University of Reading, UK
[2] System Associates Ltd, Maidenhead, UK
obinnao@systemassociates.co.uk, k.nakata@henley.reading.ac.uk

Abstract. This paper describes the implementation of a semantic web search engine on conversation styled transcripts. Our choice of data is Hansard, a publicly available conversation style transcript of parliamentary debates. The current search engine implementation on Hansard is limited to running search queries based on keywords or phrases hence lacks the ability to make semantic inferences from user queries. By making use of knowledge such as the relationship between members of parliament, constituencies, terms of office, as well as topics of debates the search results can be improved in terms of both relevance and coverage. Our contribution is not algorithmic instead we describe how we exploit a collection of external data sources, ontologies, semantic web vocabularies and named entity extraction in the analysis of underlying semantics of user queries as well as the semantic enrichment of the search index thereby improving the quality of results.

Keywords: Hansard, Named Entities, Semantic Search, RDF.

1 Introduction

Information retrieval in a number of domain centric search applications is still vastly limited to keyword based extraction techniques and the current search infrastructure for Hansard is no exception. As a consequence, the semantics and word sense behind a user's information need are not adequately captured in the search result. The importance of Hansard to the parliamentary community as a means for not just recording parliamentary debates but as an avenue for discovering trends and sourcing opinions from commentary made by members of parliament (MPs) makes its current search capability a prime candidate for semantic enhancement. Central to achieving this goal is the enrichment of the search index with contextual and semantic metadata as well as an extensive domain ontology for disambiguating topics, persons, places and events. Our approach entails extracting and indexing named entities using traditional information extraction methods [2,3], semantic metadata from documents which point to concepts in a domain ontology to support named entity (NE) extraction or conceptual disambiguation as well as the ability to make inferences about entities and

H. Takeda et al. (Eds.): JIST 2012, LNCS 7774, pp. 344–349, 2013.

instances in the domain. These features enriches and enhances the usefulness of the index and paves the way for semantically classifying commentaries and documents and finally augments the task of cross referencing content. Also, because we capture temporal and geographical information about MPs such as time in office, changes in position, party or constituency in our ontology means that geographical and temporal queries can be answered and ranked while incorporating information from external sources like the linked data space through a common URI. In the rest of this paper, we describe the characteristics of Hansard (Sec. 2), system architecture and design (Sec. 3), open source technologies used in implementing the design (Sec. 4), while sections 5 and 6 reports on the results, observations and conclusions respectively.

2 Hansard

The document search space of this semantic search is Hansard, the official proceedings of parliamentary debates, which is publicly available via web[1]. The electronic data on Hansard is semi structured and its underlying structure features the identity of a speaker followed by the commentary, within a theme or subject of debate. By transforming each Hansard document into speaker and commentary pairs, we can connect entities to external related content on the web through their URIs and analyze each speaker-commentary pair independently and in the context of sequential speaker-commentary pair embedded within a debate.

The domain specificity of Hansard to parliament means that various knowledge about aspects of the parliament and MPs are assumed, such as MPs and their constituencies, terms of office, and their changes over the years. These are often known to the users but not captured by standard keyword-based searches, which results in mismatches between user expectations and search results. The semantic search engine developed aims to improve both the relevance and coverage of search.

3 Design

3.1 System Architecture

A component based software development approach was employed for the system architecture. This offers rapid and easy development of system resources as well as its wide acceptance in the development community [1]. It consists of three tiers, data tier, analysis tier, and client tier, as illustrated in Fig. 1.

The **Data Tier** consists of two primary components. The *document harvest* component consists of a crawler that acquires all the Hansard web pages. Each document is fed to a document transformer tasked with preprocessing the documents. This involves cleaning unwanted html tags, extracting the required segments of the document and transforming it to a normalized format to enable us deal with the disparity and variation in the style of each document. The resulting transformed documents are

[1] `http://www.parliament.uk/business/publications/hansard/`: Last accessed 070/9/2012.

stored in database from which they are accessed, semantically analyzed in the analysis tier and subsequently indexed. The *document knowledge* component consists of a web data extractor that queries external data sources for parliamentary domain information ranging from personal details to time in office or positions held is organized, converted to triples and through a template in the ontology builder, marked up in RDF. We implement a parliamentary ontology and populate it with the triples after which they are stored in the Ontology Repository. The ontology is accessed through a REST API.

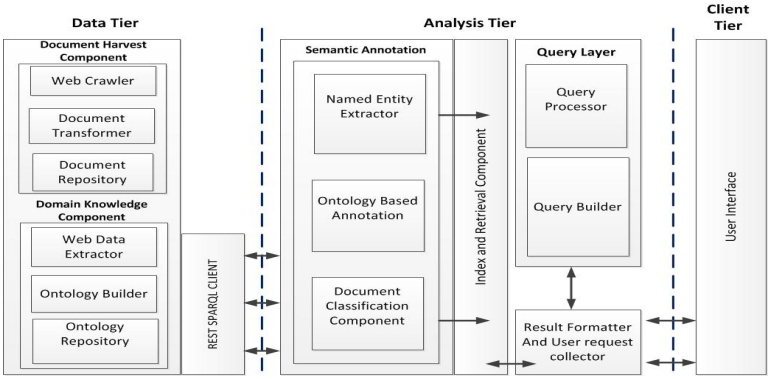

Fig. 1. System Architecture

The **Analysis Tier** encompasses the Semantic Annotation Layer (SAL), the indexing mechanism, a search query layer and request and result collector which basically serves as a bridge between the rest of the analysis tier and the client by receiving client request and returning formatted search result response to the client interface. The SAL consists of a named entity (NE) extractor that includes tokenizer, stemmer, English language POS tagger, a dictionary of parliamentary terms and a syntactic analyzer for extracting entities such as events, names, locations, organizations, dates, financial and numerical figures, dates and addresses. This rich metadata set when combined together provides the foundation for semantic and contextual relevance in the result set. We enhance the metadata derived from the NE extractor by employing the ontology based annotation component in extracting relevant annotations which point to concepts in the domain ontology. The hierarchical structure of the ontology means that we can disambiguate extracted NEs and make inferences. The document classification model is based on the assumption that a set of words occurring together will determine its sense [4, 5] so through the existing classification of concepts and entities in domain ontology we have a credible base from which document classification can be performed by analyzing the document similarity based on extracted NEs.

All extracted metadata and categorization information is indexed and queried via the query layer. Query layer comprises of a query builder and processor. The processor also harnesses NE tools in extracting the key components of the user requests. For instance a query like “How much was spent by the Labour Government on Health in

1998?" The syntactic analyzer would classify the query type as a question. Relations among the entities are derived from the query, i.e., a relationship "on" is formed between "Labour Government" (Type: Organization) and "Health" (Type: Topic). The temporal assertion in the request with the year "1998" identified by the NE extractor provides a temporal constraint. In addition, through the domain ontology, the entities "Labour Government" and "Health" are associated with related entities. These parameters forms a query structure based on which relevant documents are searched.

3.2 Process Flow

There are two primary processes involved in the system (Fig. 2). **Data collection and semantic indexing** (dotted arrows in) is a non-real time scheduled process primarily to support updates to ontology and Hansard documents. **User queries** (solid arrows) trigger a search process. The data collection and semantic indexing process commences with the crawling, transformation and population of transformed documents into the database repository. Domain information is sourced from the web, marked up in RDF and used to populate the ontology repository. The completion of these processes triggers the task of analyzing and extracting named entities from the documents which generates indexes based on semantic metadata. User queries are first processed by the query processor to derive a set of query terms which include query entities, query context, relations in the query and disambiguation of entities. The resulting query parameters are converted to a structured query by the query builder which then queries the index, which returns the results to a result formatter which forwards it to the client interface.

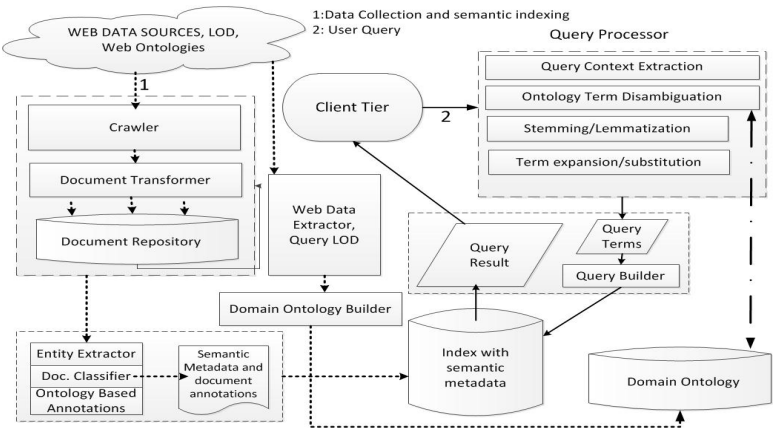

Fig. 2. Process Flows within the System

4 Implementation

In the current implementation, the system crawled Hansard documents over a five year period. Our objective was to obtain an initial sizable corpus that captures periods

covering notable transitions in parliament through elections, hand-overs, resignations etc. Preprocessing these documents involved stripping them of unwanted html tags and extracting speaker-commentary pairs and valuable metadata like the original document's URL, date and topic if available. Through this speaker-commentary pairs and other relevant document metadata are obtained. Mongo[2] DB was used in housing the transformed document and its metadata. The choice of Mongo is due to its document oriented structure, scalability, JSON like data structure and popularity. The domain ontology was implemented using Bigdata, an open source semantic web repository which supports SPARQL queries and RDFS inferences. Domain ontology data is sourced primarily from "theyworkforyou.co.uk" web site which maintains a REST API for querying MP and parliamentary information. The resulting parliamentary ontology makes use of popular semantic web vocabularies like FOAF, SKOS and Dublin Core. The data from "theyworkforyou.co.uk" is converted to triples and stored in the ontology. The resulting ontology contains about 2 million triples. NE extraction is carried out using GATE[3], an open source text engineering tool by modifying its Gazetteer and creating new grammar rules to identify and extract additional context like questions, responses to questions, statements. For our topic categorization, we make use of Open Calais[4]'s topic categorization and scoring mechanism through its public API. Commentaries are therefore classified according to a genre such as politics, social issues etc. The resultant entities are indexed in Solr[5], an open source search platform based on Lucene. At query time, terms in the user query are expanded through the domain ontology and disambiguated in the way that we can indicate that a search for the term "Maidenhead" refers to the constituency of Maidenhead, which has as its MP Theresa May. We can also make temporal disambiguation such that Prime Minister between 2010 and 2012 is David Cameron while Prime Minister in 2009 was Gordon Brown, the MP for Kirkcaldy & Cowdenbeath. This sort of disambiguation extends to names, titles and positions.

5 Results and Observations

Using independent assessors, early stage evaluation of our search engine in comparison with the current one focused on relevance of the results. Testers judged and recorded degree of relevance for the first 15 results as "Relevant", "Quite relevant" and "Not relevant" for 6 different search phrases. On average, 32% of the results in our search were classified as very relevant as compared to 11.6% in the current system. Our result also exhibits salient features of semantic expansion. For example, a search on "What David Cameron has said about Health Care between 2001 and 2011" returns two sets of results. The first set features statements made by David Cameron, the MP for Witney prior to his appointment as Prime Minister in 2010, while the second set includes statements made by the Prime Minister David Cameron. Fig. 3 is a screenshot of the outputs from the demonstrator.

[2] `http://www.mongodb.org/`: Last accessed 07/09/2012.
[3] `http://Gate.ac.uk`: Last accessed 07/09/2012.
[4] `http://Opencalais.com`: Last accessed 07/09/2012.
[5] `http://lucene.apache.org/solr/`: Last accessed 07/09/2012.

The combination of domain ontology, the indexing of named entities and contextual information means that we can compose more expansible yet focused queries which fit the user's information need. Our component based approach also means that components can be replaced or substituted. For instance, we can enhance the strength of our domain ontology and disambiguation process by directing queries to Dbpedia to obtain more extensive information on entities. The NE extraction process is not always accurate and there are situations where we are unable to identify some named entities or even misidentify certain named entities.

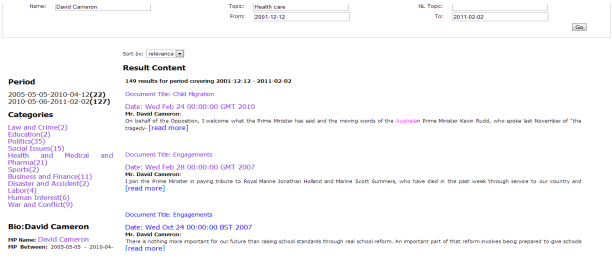

Fig. 3. A screenshot of search results

6 Conclusion and Future Work

In this paper we have shown that a hybrid approach combining NE extraction resources and domain ontology can serve to implement semantic search for Hansard. We intend to further enrich the domain ontology by accessing additional data sources. Future directions include extending the semantic extract sentiments and opinion in the comments made by MPs so that search results can be enriched by offering both single and collective sentiments on debated subject matters.

References

1. Emmerich, W.W.: Distributed Component Technologies and their Software Engineering Implications. In: Proceedings of the 24th International Conference on Software Engineering, Orlando, Florida, pp. 537–546 (2002)
2. Junhui, Y., Chan, H.: Keywords Weights Improvement and Application of Information Extraction. In: Gaol, F.L., Nguyen, Q.V. (eds.) Proc. of the 2011 2nd International Congress on CACS. AISC, vol. 144, pp. 95–100. Springer, Heidelberg (2012)
3. Lam, M.I., Gong, Z., Muyeba, M.K.: A Method for Web Information Extraction. In: Zhang, Y., Yu, G., Bertino, E., Xu, G. (eds.) APWeb 2008. LNCS, vol. 4976, pp. 383–394. Springer, Heidelberg (2008)
4. Liu, Y., Scheuermann, P., Li, X., Zhu, X.: Using WordNet to Disambiguate Word Senses for Text Classification. In: Shi, Y., van Albada, G.D., Dongarra, J., Sloot, P.M.A. (eds.) ICCS 2007, Part III. LNCS, vol. 4489, pp. 781–789. Springer, Heidelberg (2007)
5. Navigli, R.: Word Sense Disambiguation: A Survey. ACM Computing Surveys 41(2), Article No. 10 (February 2009)