GATE Mímir:
Answering Questions Google Can't

Mark A. Greenwood, Valentin Tablan, and Diana Maynard

Department of Computer Science,
University of Sheffield, UK
initial.surname@dcs.shef.ac.uk

Abstract. Free text makes up a large proportion of the vast amounts of information generated by modern society, and search engines such as Google are exceptionally good at finding, indexing and searching this. However, the rise of the Semantic Web and the publishing of increasingly large amounts of structured and interlinked data now means that useful information is distributed across multiple sources and in a variety of formats, which cannot be easily reconciled by these search engines as it is not amenable to free text search. Hence, questions which we may wish to ask of society's collective knowledge cannot be easily answered. For example, it is difficult to see how traditional search engines could be used to locate documents in which a person born in Sheffield is being quoted. In this paper, we describe GATE Mímir which indexes not only free text, but also semantic annotations and knowledge base data. The resulting multi-paradigm index allows us to search across multiple information sources in order to answer questions which are either infeasible or impossible to answer using current web search engines.

1 GATE Mímir

GATE Mímir[1] is a multi-paradigm information management index and repository which can be used to index and search over text, annotations, semantic schemas (ontologies), and semantic metadata (instance data). It allows queries that arbitrarily mix full-text, structural, linguistic and semantic queries, and that can scale to gigabytes of text. We briefly describe in this paper the underlying architecture; full details can be found in [1].

A GATE Mímir index supports combinations of the following data types:

Text: Support for full text search represents the most basic indexing functionality, and it is required in most (if not all) cases. In GATE Mímir this is implemented as an MG4J[2] inverted index.

Annotations: The first step in abstracting away from plain text document content is the production of annotations. Annotations in this case are linguistic metadata associated with text snippets in the documents. For example, if the

[1] Open source and available via http://gate.ac.uk/family/mimir.html
[2] http://mg4j.dsi.unimi.it/

documents are annotated with occurrences of person and organization entities, then searches such as {`Person`} `", CEO of"` {`Organization`} become possible (where {`Person`} and {`Organization`} can represent any person and organisation's name in the text respectively, and `", CEO of"` is a textual string). So for example, the query would return a document containing the string "Ian Cheshire, CEO of Kingfisher plc". Exact string matching is not required necessarily: we can also use wildcards, standard binary operators, and different linguistic forms of words (e.g. `root:say` will match any form of the verb "say" such as "saying", "says", "said" etc.).

Knowledge Base Data: This consists of an ontology, representing the data schema, populated with instances. The instance data represents facts that are known to the system and are typically derived (at least partially) from the semantic annotation of documents. A Knowledge Base (KB) pre-populated with appropriate world knowledge can perform generalisations that are natural to humans, such as knowing that Vienna is a valid answer to queries relating to a city in Austria or Europe. While a GATE Mímir index can contain KB data, it is often more useful to link an index with one or more external repositories via public SPARQL endpoints.

2 Building a GATE Mímir Index

The example index presented in the remainder of this paper includes all three data types described above. The demo index was built from 8,255 articles downloaded from the BBC News website during April 2011. The documents were then annotated with a GATE application based on ANNIE [2]. We customized the way in which ANNIE annotates the names of people, organizations and locations, using the Large Knowledge Base (LKB) gazetteer, which enables us to annotate these concepts directly from a semantic repository, rather than from a predetermined and flat set of gazetteer lists. This should lead to greater coverage and better precision; more importantly, however, it means that certain annotated entities are linked to specific instance URIs in a semantic repository. The LKB instance we use in this application is loaded from DBpedia[3]. A number of processing resources to determine the date, title and classification of each article were also added to the application.

The annotation of all 8,255 articles and the generation of the GATE Mímir index was completed in just 36 minutes on a single desktop PC[4].

It is worth noting that in this example the KB is actually used both during the annotation of the documents and at search time, although the two uses differ substantially. As mentioned above, a gazetteer is created locally from a KB which uses instance labels to annotate documents, with each annotation that is created including the instance URI. At search time, those same instance URIs can be used within a SPARQL query to search for related information which can then be used to restrict the search. In this example, the GATE Mímir

[3] `http://dbpedia.org`

[4] 64bit Ubuntu running on an Intel Core 2 Quad 2.66GHz with 8GB of RAM

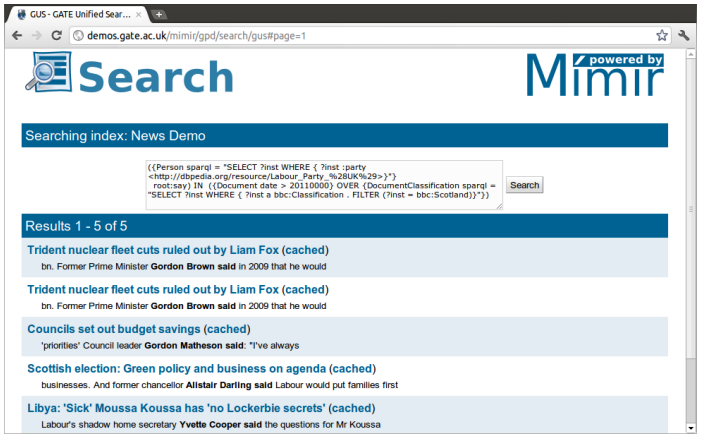

Fig. 1: GUS, The GATE Mímir Query Development Interface `http://demos.gate.ac.uk/mimir/gpd/search/gus`

index does not actually contain a KB, rather it was configured to point to a Sesame server[5] containing both DBpedia and a small ontology describing the classification schema of the indexed BBC News articles.

3 Multi-Paradigm Search

The main advantage of GATE Mímir over existing search engines is that there are interesting questions which only GATE Mímir is able to answer, e.g. find all documents in which a person born in Sheffield is quoted. The combination of text, annotations and KB data in the index allows us to find such documents using the following query:

```
{Person sparql = "SELECT ?inst WHERE{?inst :birthPlace
<http://dbpedia.org/resource/Sheffield>}"} [0..4] root:say
```

This query basically says: *find all the places at which a Person annotation, whose URI (stored in the inst feature) is mentioned in DBpedia as being born in Sheffield, occurs within at most five words of the verb to say.* It is important to note that none of the documents which match this query mention place of birth. This nicely highlights the power of combining annotations, free text, and data from a KB when searching a document collection. GATE Mímir comes with a simple query development interface, known as GUS (Figure 1), where this example and other queries can be experimented with.

The GATE Mímir query syntax is very powerful, but can easily result in very complex queries that no one is likely to want to enter by hand. In other words (and with apologies to George Lucas[6]), GUS is (probably) not the interface you are looking for. Fortunately, GATE Mímir also supports a RESTful XML-based search interface which can be used to build custom search interfaces quickly. Custom interfaces are likely to focus on one aspect of an index, and multiple

[5] `http://www.openrdf.org/`
[6] `http://www.imdb.com/title/tt0076759/quotes?qt=qt0440731`

Fig. 2: A Custom GATE Mímir Interface
`http://demos.gate.ac.uk/pin/`

interfaces could easily be developed for the same index. An example can be seen in Figure 2: this focuses solely on finding people in the indexed documents. The search can be restricted to aspects of both the person (their name, where they were born and what they are famous for) as well as the documents in which they occur (the date of the document, the BBC classification and if boilerplate text around articles should be ignored). This simple interface allows for quite complex queries to be built rapidly and results returned without the user having to know anything about SPARQL or the GATE Mímir query syntax.

In addition to the demo system presented here, GATE Mímir has also been used in two real life applications: one in the Patents domain[1] and one for The National Archives[7]. The former system does not involve semantic queries, and was used for a corpus of around 20 million documents. The latter includes semantic querying and runs over approximately 150 million documents (the entire public collection of unique online documents from the National Archives). Because of the huge data size, the annotation and indexing for this was achieved using the GATE Cloud Parallelizer[8] installed on the Amazon cloud. Demos for all three applications, as well as the custom "People in the News" interface, can be found at `http://demos.gate.ac.uk/mimir/`.

References

1. Cunningham, H., Tablan, V., Roberts, I., Greenwood, M.A., Aswani, N.: Information Extraction and Semantic Annotation for Multi-Paradigm Information Management. In Lupu, M., Mayer, K., Tait, J., Trippe, A.J., eds.: Current Challenges in Patent Information Retrieval. Volume 29 of The Information Retrieval Series. Springer (2011)
2. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. In: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02). (2002)

[7] `http://www.nationalarchives.gov.uk`
[8] `https://gatecloud.net`