AMBIT-SE: Towards a User-aware Semantic Enterprise Search Engine

Giacomo Cabri, Stefano Gaddi and Riccardo Martoglia
University of Modena and Reggio Emilia, Modena, Italy

Keywords: User-awareness, Enterprise Search Engine, Information Retrieval, Text Analysis, Semantic Knowledge and Similarity.

Abstract: Search engines represent one of the most exploited tools both in our everyday life and in our work. In this paper we propose a user-aware semantic enterprise search engine called AMBIT-SE. It is "enterprise" in the sense that it is focused on the search in enterprise websites; the "semantic" aspect is related to the fact that it exploits not an exact word match, but relies also on the meaning of the words by means of synonyms and related terms; finally, to produce query results it takes into account also the user information, which turns out to be very useful to improve the search. We explain how our system works and report the results of experiments on different websites.

1 INTRODUCTION

In today's enterprises the need for providing appropriate means to search for specific information in internal and public repositories is more and more increasing. This goal is twofold: from the one hand, enabling employees to find the needed information in a short time is not only useful to reduce the global time need to carry out a task, but also to decrease the frustration of long searches; on the other hand, precise and relevant answers to customers that exploit the company web sites for both searching for information and interacting with the company can grant a high degree of customer satisfaction.

In this context, two aspects that can improve the relevance of the search results are *semantics* (Mangold, 2007) and *user-awareness* (Xiang et al., 2010). Semantics can be useful to overcome the limitations of a syntactic approach, which is often exploited but leads to a reduced number of results. User-awareness can be useful to tailor the search results on the base of the context of the user that performs a query or a request. As far as we know, there are no enterprise search engine that exploits both aspects in a single approach.

Starting from this consideration, this paper proposes a user-aware semantic enterprise search engine that was built with this goal in mind, describing in detail its architecture and how it works. The search engine is called AMBIT-SE (AMBIT Search Engine). It is not a generic search engine, but a search engine dedicated to an enterprise website. We exploit semantic techniques to improve the search: instead of a pure syntactic matching between the query keywords and the words in the available documents, we rely on their meaning and take into account synonyms and related terms. Moreover, the main innovation of our approach is to exploit user information to further improve the search results. In fact, the approach we propose takes advantage of textual information, certainly the primary component of the documents that should be presented / suggested to users, and also one of the main information characterizing user profiles (think, for instance, to the contents of user browsing history, to the description of users' interests, and so on).

Our innovative approach is based on text analysis, semantic retrieval and user-aware techniques and leads to the following achievements:

- its semantic features are powerful enough to provide enhanced searching effectiveness over standard search techniques;
- thanks to user awareness, search results actually reflect the user's preferences and needs;
- it is general, flexible and able to process multilingual information;
- it is devised for IT SMEs, providing them with easy-to-apply methods that do not require big investments or knowledge prerequisites, allowing them to query for the information they need in the way they are used to.

Cabri, G., Gaddi, S. and Martoglia, R.
AMBIT-SE: Towards a User-aware Semantic Enterprise Search Engine.
In *Proceedings of the 12th International Conference on Web Information Systems and Technologies (WEBIST 2016) - Volume 2*, pages 98-108
ISBN: 978-989-758-186-1

This paper is organized as follows. First, we present an overview of the proposed search engine (Section 2). Then, we explain how our approach analyzes the documents that can be "searchable" by the users (Section 3), and how it defines which documents must be retrieved to satisfy the user's query (Section 4). We report the results of the experiments carried out on our system (Section 5). Finally, before the conclusions (Section 7) we report some related work (Section 6).

2 SYSTEM OVERVIEW

In this section we present an overview of the proposed semantic enterprise search engine. Figure 1 proposes the workflow of AMBIT-SE, which is composed by a series of coordinated offline and online processes.

First of all, AMBIT-SE performs a *Document analysis* phase, in which any textual information available in the documents that will need to be retrieved (e.g. web pages for a given site) and in the documents useful to determine the user's behaviour and preferences (such as e-mails, web pages viewed, profile information, past search queries, etc.) is extracted and processed. The workflow starts by using a Web Crawler to retrieve the raw data of all the documents that must be searchable, such as the web pages belonging to a portal. All of these files are then submitted to the actual analysis process, which consists of the following steps:

1. The textual content of all files is extracted;
2. The language of each file is determined;
3. The content is divided into paragraphs, and each paragraph is divided into lines of text;
4. Each line of text is divided into "Tokens" (single terms);
5. The "Stem" (dictionary form of a term) and "Part of speech" value (basic type of term) of each token is determined;
6. Terms classified as nouns are preserved;
7. Nouns are processed with word sense disambiguation algorithms, in order to be able to compute synonyms and related terms information from a thesaurus;
8. The weight of each noun, corresponding to its containing document, is calculated.

The data structure containing all the document analysis results will be referred to as "Website(s) semantic glossary". At the same time, the documents that constitute the user's profile, such as all of the web pages he has visited, are analyzed in the same way, resulting in a "User semantic glossary". Both glossaries are then compared with document similarity algorithms (see "Semantic glossaries computation and comparison" in the figure), and a "Profile ranking" is determined, symbolizing how relevant the retrievable documents are in relation to the user's preferences.

The AMBIT-SE online phase allows users to search through the retrievable data index with different kinds of queries, resulting in a "Query ranking". The two rankings are then normalized and merged, so that the final ranking of the retrieved files will take into account both the query relevance and the user's preferences based on the results of the aforementioned process.

The software is written in Java and exploits several Open-source programs and libraries. The currently supported languages for all operations are: English, Italian, Spanish, German, French, Finnish, Dutch, Polish.

The following sections (Section 3 for document analysis and Section 4 for document retrieval) provide in-depth information on each step of the text processing pipeline, illustrating the theory behind them and the techniques used to execute them.

3 DOCUMENT ANALYSIS

3.1 Crawling

At the beginning of the crawling process, the "Web crawling" module creates a document data store ("Raw document data" in Figure 1) containing the raw data of the documents which will be analyzed by the document analysis steps together with the details about their source. In order to extract retrievable document data we employ different type of crawlers:

- The *Web Crawler* handles HTTP and HTTPS, and is used to crawl Internet, intranet and extranet sites;
- The *File Crawler* handles local or remote file systems; It can retrieve local document data by crawling the local file system and the NFS and CIFS mount points, and remote document data using the following protocols: CIFS/SMB, FTP, FTPS.

The raw document data store contains, among the others, the following fields for each of the documents: Title, Content, URL, File Name, Meta Description, Meta Keywords, Host name, Subdomain, Backlink Count (i.e. number of incoming links). All of the crawling operations are handled by the Open-source

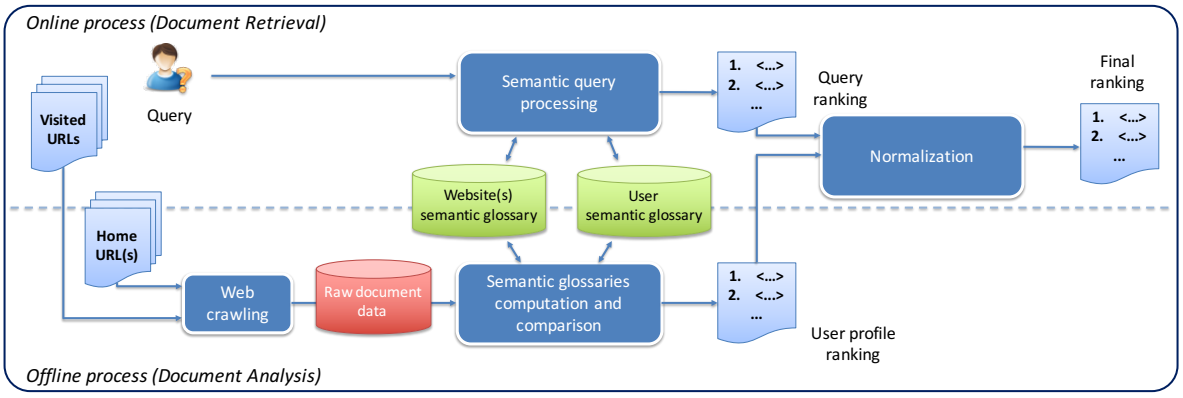

Figure 1: The main processes of the AMBIT-SE user-aware semantic enterprise search engine.

enterprise class search engine software, OpenSearchServer[1].

3.2 Extraction, Language and Paragraphs

In this further step the text contained within each document needs to be extracted, an operation that can vary greatly depending on the format of each document; for instance, extracting text from an HTML file implies excluding every tag, script and comment within. Next, AMBIT-SE determines the language of each file, because there are slight variations in the workflow based on it. Then, the extracted content of each file is divided into paragraphs, and each paragraph into lines of text; this will allow the final ranking to show not only which documents the user is most interested in, but also which paragraphs and lines within have affected this result the most. In particular, the system will show a text snippet of the highest rated paragraph when presenting a document to the user.

All of these preliminary operations are taken care of by components of the Open-source software GATE[2].

3.3 Tokenization, Stemming and Parts of Speech

The next analysis step is tokenization, i.e. the process of breaking the stream of text up into terms, phrases, symbols, or other meaningful elements called tokens. In this case, it is performed by a language-independent Java function that implements methods for finding the location of boundaries in text, and then splits it accordingly in order to divide it into single terms and symbols.

The list of tokens becomes input for the stemming, which is the process of determining the dictionary form, called stem, for a given term. Stemming is language-dependent; in AMBIT-SE we support 17 languages: English, German, Arabic, Chinese, Danish, Spanish, Finnish, French, Dutch, Hungarian, Italian, Norwegian, Portuguese, Romanian, Russian, Swedish, Turkish. Also, stopwords are removed.

In addition, the tokens are subjected to POS tagging, where each term is marked as corresponding to a particular Part Of Speech. Simply put, the tagger identifies terms as nouns, verbs, adjectives, adverbs, etc. Only the terms classified as nouns are considered relevant, and will thus be preserved for the rest of the procedure.

In this case, both stemming and POS tagging are taken care of by TreeTagger[3], a tool for annotating text with part-of-speech and lemma information. To make use of TreeTagger, the TT4J[4] (TreeTagger for Java) Open-source wrapper is employed.

3.4 Term Weights

As in classic Information Retrieval (Baeza-Yates and Ribeiro-Neto, 1999) the importance (weight) of each term t in each document D of the document collection $\mathcal{D}$ is estimated. We exploit tf-idf weighting, which is the product of two statistics:

- *TF: Term Frequency,* which measures how frequently a term occurs in a document. Since every document D is different in length, it is possible that a term t would appear much more times in

[1]http://www.opensearchserver.com/
[2]https://gate.ac.uk/
[3]http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
[4]https://reckart.github.io/tt4j/

long documents than in shorter ones. Thus, the term frequency is divided by the document length as a way of normalization:

$$\mathrm{tf}(t,D) = \frac{\mathrm{f}(t,D)}{\mathrm{len}(D)}, \quad (1)$$

where $\mathrm{f}(t,D)$ is the raw frequency of the term t in the document D (number of times the term appears in the document), and $\mathrm{len}(D)$ is the total number of terms in the document D;

- *IDF: Inverse Document Frequency,* which measures how important a term is. While computing TF, all terms are considered equally important. However it is known that certain terms may appear often but have little importance. Thus we weigh down the frequent terms while scaling up the rare ones, by computing the following:

$$\mathrm{idf}(t,\mathcal{D}) = \log \frac{N}{|\{D \in \mathcal{D} : t \in D\}|}, \quad (2)$$

where N is the total number of unique documents (among both the user's and retrievable data collections), and $|\{D \in \mathcal{D} : t \in D\}|$ is the number of documents where the term t appears.

Then, tf-idf is calculated as:

$$\mathrm{tfidf}(t,D,\mathcal{D}) = \mathrm{tf}(t,D) \cdot \mathrm{idf}(t,\mathcal{D}) \quad (3)$$

3.5 Semantic Analysis and Thesauri

One of the main features of AMBIT-SE is the ability to exploit the semantics of the text, going beyond standard syntactical search engines. In particular, we want to extend the search to synonyms and related terms of a given term. To handle this, we exploit WordNet[5], a large lexical database where nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (*synsets*), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations, resulting in a network or meaningfully related words and concepts. WordNet superficially resembles a thesaurus, in that it groups words together based on their meanings. However, WordNet interlinks not just word forms, but specific senses of words. Therefore, AMBIT-SE first performs Word Sense Disambiguation (WSD) of the text.

Before discussing WSD, let us briefly discuss the structure of WordNet in order to explain how knowing the synset(s) associated with a text will allow to easily compute its synonyms and related terms. Synonyms

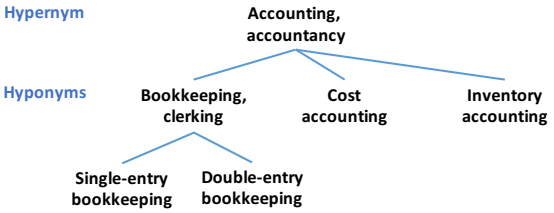

Figure 2: Hypernym/hyponym hierarchy example.

are basically terms associated to the same synset; related terms are typically represented by their *hypernyms* and *hyponyms*. In linguistics, a hyponym is a term whose semantic field is included within that of another term, its hypernym. In simpler terms, a hyponym shares a type of relationship with its hypernym; on the other hand, a hyponym is a term whose semantic field is more specific than its hypernym. The semantic field of a hypernym, also known as a superordinate, is broader than that of a hyponym. Figure 2 shows a small example: the terms "bookkeeping" and "clerking" are synonyms, i.e. they belong to the same synset. Terms "accounting" and "accountancy" constitute their hypernym, while possible hyponyms are "single-entry bookkeeping" and "double-entry bookkeeping". By expliting this information, we will allow users looking for "accounting" information to easily retrieve documents containing different but related words like "bookkeeping".

Please also note that, while the original WordNet is strictly in English, our goal was to provide multilingual semantic coverage. To this end, we also exploit the custom WordNet versions available for the different languages that have been collected, extracted and normalized in the Open Multilingual WordNet[6] project. In addition, we exploit the automatically extracted data from Wiktionary and the Unicode Common Locale Data Repository. This allows comparisons between different languages, as terms are represented by corresponding WordNet synset codes.

Getting back to WSD, to identify which sense of word (i.e. meaning) is used in a sentence, the exploited algorithm evaluates the similarity between the synsets a of each term to be disambiguated and the synsets b of the other nearby terms. The (Leacock and Chodorow, 1998) metric is used: this measure relies on the length of the shortest path between two synsets for their measure of similarity. We limit our attention to hyponymy/hypernymy links and scale the path length by the overall depth T of the taxonomy.

Path length similarity between synset a and synset b is computed using the formula:

$$sim_{ab} = \max_p [-\log(N_p/2T)], \quad (4)$$

[5]https://wordnet.princeton.edu/

[6]http://compling.hss.ntu.edu.sg/omw/

Table 1: A small excerpt of semantic glossary (per-document view).

Document	Term	Synsets	tf	Weight(tf*idf)
OP0001	bookkeeping	00619230-n	0.333	0.135
OP0005	accounting	00618734-n, 13354985-n	1	1.098
...	...	...	...	...

where N_p is the number of nodes in path p from a to b, and T is the maximum depth of the taxonomy.

The result of the process is a disambiguation score *dis* assigned to each synset code belonging to each term: it is normalized between 0 and 1, and it represents the chances of that term having that sense in that context. Only the synsets whose disambiguation score exceeds a given threshold th_d, i.e. $dis > th_d$, will be kept and stored as the result of the analysis.

3.6 Semantic Glossaries

The result of the analysis of the documents is stored in a structure we call semantic glossary. In particular, the analysis of all the retrievable documents in the collection is stored in the "Website(s) semantic glossary", while the analysis of the documents associated with the user profile (i.e. visited URLs, etc.) is stored in the "User semantic glossary"; the two glossaries share the same structure. Each glossary is composed of two "views" which store:

- all the terms (and their synsets) in the documents with their statistics (*global view*);
- the terms occurrences (and their synsets) in each document with their statistics (*per-document view*).

In particular, the glossary global view is an alphabetical sort of all the extracted terms, while the glossary per-document view is a list of all the term occurrences in the documents, sorted by the document ID, together with their statistics. A small excerpt of a glossary per-document view is shown in Table 1: the columns include the document ID ("Document"), the contained term ("Term"), the WordNet synset code(s) ("Synsets") as derived from WSD, and the term frequency ("tf") and tf-idf weights as described in Section 3.4.

As we will see in the next section devoted to document retrieval, by means of the stored synset and weight information, the content of the glossary allows the similarity functions of AMBIT-SE to draw useful knowledge from both the semantic (i.e. synonyms and related terms computation) and the text retrieval fields.

4 DOCUMENT RETRIEVAL

The final goal of the document retrieval process in AMBIT-SE is to effectively answer a given query Q submitted by a user U; to this end, it takes into account all the semantic and user profile information available in the semantic glossaries produced by the analysis process and generates a ranking of the available documents.

The computation of the document ranking is based on ad-hoc similarity metrics:

- the similarity between the main terms of the available documents and those specified in the query;
- the similarity between the documents' terms and those associated with the user profile (e.g. past navigated documents).

Both similarities are based on a general document similarity formula which we will detail in the following section; finally, in Section 4.2 we will analyze some further aspects regarding AMBIT-SE query processing.

4.1 Document Similarity Computation

Building on previous research on text retrieval for specific subject areas as software engineering (Bergamaschi et al., 2015; Martoglia, 2011), bibliographical (Beneventano et al., 2015) and user-centric data (Martoglia, 2015), we define the following document similarity formula:

$$DSim(D^x, D^y) = \sum_{t_i^x \in D^x} TSim(t_i^x, t^y_{\bar{j}(i)}) \cdot w_i^x \cdot w^y_{\bar{j}(i)}, \quad (5)$$

where:

$$t^y_{\bar{j}(i)} = argmax_{t_j^y \in D^y}(TSim(t_i^x, t_j^y)),$$

$$w_i^x = tf_i^x \cdot idf_i,$$

$$w^y_{\bar{j}(i)} = tf^y_{\bar{j}(i)} \cdot idf_{\bar{j}(i)}$$

and $TSim$ is a term similarity formula (see following) taking into account the semantic information extracted from the semantic glossary. Simply put, the similarity $DSim(D^x, D^y)$ between two documents D^x and D^y is determined by summing the maximum term similarity score $TSim(t_i^x, t^y_{\bar{j}(i)})$ between each pair of terms belonging to different documents, multiplied by the tf-idf weights of both.

We now proceed to define $TSim(t_i, t_j)$ between two terms t_i and t_j. In our semantic framework, t_i and t_j can be:

- *Synonyms*, i.e. t_i SYN t_j, if a common synset a is stored in the semantic glossary for both terms, i.e. $\exists a \in t_i \cap t_y$ (note that this case includes equal terms);

- *Related*, i.e. t_i REL t_j, if the synset similarity sim_{ab} between two synsets $a \in t_i$ and $b \in t_j$ (Eq. 4) exceeds a given threshold th_s, i.e. $\exists a \in t_i, b \in t_j | sim_{ab} > th_s$;
- *Unrelated*, otherwise.

The corresponding term similarity scores are assigned as follows:

$$TSim(t_i,t_j) = \begin{cases} 1, & \text{if } t_i \text{ SYN } t_j \\ r, & \text{if } t_i \text{ REL } t_j \\ 0, & \text{otherwise} \end{cases}, \quad (6)$$

where r is an arbitrary value between 0 and 1. Please note that synonym and related term information can be computed offline for all the documents in the collection and in the user profile.

By applying Eq. 5 to the query Q and to each retrievable document D_x of the user profile U, we obtain a "query ranking" and a "user profile ranking", respectively, of retrievable documents D_y in the collection. The two rankings are then normalized and merged in a final ranking taking into account both the user's request and preferences.

4.2 Further Query Processing Aspects

Besides the techniques and features described in detail in the previous sections, AMBIT-SE also offers administrators the following ways to customize query processing and presentation of results:

- *Text Snippets,* showing a custom-sized highest rated portion of each document in the presented ranking, plus any amount of surrounding text necessary to reach the desired size; also, the tag used to highlight words in results listings is parametrable;
- *Boosting Subqueries,* to tweak the relevance score of documents. For instance, a website administrator could specify certain group of words, i.e. those describing a new product, whose weight will be promoted when found in both the user query and a given retrievable document. The subqueries can both bolster or lower a document's score;
- *Autocomplete,* offering the most relevant suggestions to the user on the basis of the semantic glossary terms.

5 EXPERIMENTAL EVALUATION

This section illustrates and analyzes the results of several tests performed on different kinds of websites.

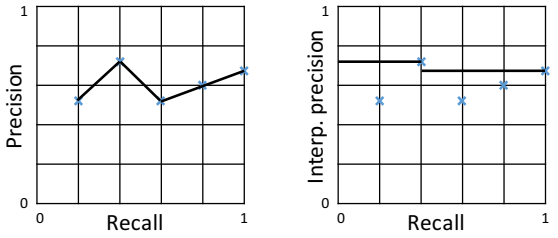

Figure 3: Effects of interpolated precision on the P-R curve.

Since the description of the underlying index structures supporting semantic search is outside of the scope of this paper, we will focus on effectiveness analysis; anyway, please note that the current prototype has a response time of 40 ms on average on a standard single-node configuration.

5.1 Ranked Evaluation Method

The measures used for evaluation are precision and recall, turned into measures of ranked lists by computing them for each top k set of results, obtaining a precision-recall curve (Baeza-Yates and Ribeiro-Neto, 1999).

In particular, we compute the interpolated precision as:

$$P_{interp}(r) = max_{r' \geq r} P(r') \quad (7)$$

The interpolated precision at a certain recall level r is defined as the highest precision found for any recall level $r' \geq r$ (see Figure 3).

The rationale for interpolation is that the user is willing to look at more records if both precision and recall get better. The interpolated precision is measured at 11 recall levels of 0.0, 0.1, 0.2, ..., 1.0 (0, 10, 20, ..., 100 percent), then the arithmetic mean of the obtained values is calculated (Bernardi, 2011).

5.2 Experimental Setting

Four heterogenous business-relevant websites were selected for evaluation purposes; for each one of them, an appropriate information need was established by examining common searches performed in the past.

- *http://www.cobat.it/*, a relatively small Italian website that provides information and services for disposing and recycling four problematic waste categories: batteries and accumulators, tires, electric and electronic devices, and photovoltaic panels. The established information need is to retrieve documents pertaining to the disposal of batteries and accumulators.

Query	Our results							Google
	Base	Stem	Stem + SynRels	Stem + Prof	Stem + Prof + SynRels	Stem + HetProf	Stem + HetProf + SynRels	
Q1	0,54529	0,53914	0,55451	0,63225	0,80148	0,56444	0,75783	0,53006
Q2	0,04242	0,31009	0,12454	0,38424	0,47474	0,35351	0,29737	0,21818
Q3	0,32958	0,32906	0,48566	0,43939	0,80014	0,45455	0,73826	0,39610
Q4	0,00000	0,10909	0,12542	0,24242	0,39882	0,24242	0,32553	0,06061
Q5	0,08684	0,08678	0,13376	0,38678	0,46167	0,34444	0,24239	0,08392
Average	0,20083	0,27483	0,28478	0,41702	0,58737	0,39187	0,47227	0,25777

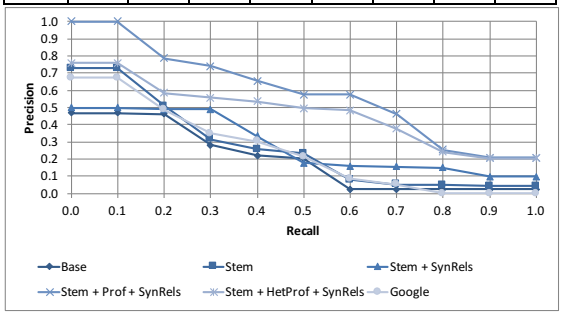

Figure 4: Test results for http://www.cobat.it/.

Query	Our results							Google
	Base	Stem	Stem + SynRels	Stem + Prof	Stem + Prof + SynRels	Stem + HetProf	Stem + HetProf + SynRels	
Q1	0,28400	0,28003	0,37134	0,53033	0,56539	0,47710	0,51782	0,51363
Q2	0,00000	0,11039	0,32255	0,06818	0,49271	0,07025	0,38474	0,09091
Q3	0,36519	0,41652	0,37863	0,45101	0,50202	0,40363	0,45740	0,45990
Q4	0,09091	0,09091	0,35949	0,09091	0,55438	0,09091	0,42646	0,15909
Q5	0,00000	0,01653	0,27722	0,06061	0,41804	0,02597	0,36979	0,00000
Average	0,14802	0,18287	0,34185	0,24021	0,50651	0,21357	0,45124	0,24471

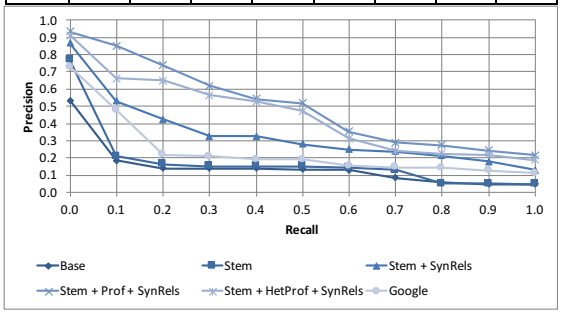

Figure 5: Test results for http://evergreensmallbusiness.com/.

- *http://evergreensmallbusiness.com/*, an English Blog that publishes different kinds of information and advice for small businesses, all classified in categories such as business taxes, management, personal finance, etc. The established information need is to retrieve articles pertaining to bookkeeping.
- *http://truegoods.com/*, an English Indie online shop that specializes on healthy and natural products. The established information need is to retrieve information on products belonging to the pet-care category.
- *http://www.gruppozatti.it/*, an Italian authorized car dealer which sells several brands of both new and used cars. The established information need is to retrieve different information about cars belonging to the used category.

For each information need, several plausible queries were submitted to the system (we selected five representative ones for this evaluation). We employ different setups in order to evaluate the impact of the different features of AMBIT-SE, as described below:

- *Base:* baseline setting, i.e. simple syntactical search for exact keywords;
- *Stem:* Stemming and stopword removal are performed;
- *SynRels:* Synonyms and related terms are also taken into account, both for the query and when processing the user profile documents, if present;
- *Prof:* a User Profile containing only documents relevant to the information need is used to perform searches;
- *HetProf:* a Heterogeneous User Profile containing documents relevant to the information need and an equal number of irrelevant documents is used to perform searches;
- *Google:* queries are run through the Google search engine restricted to the considered document set, for reference.

Please note that the first baseline is also representative of the document retrieval techniques commonly exploited by most commercial systems (see also related works).

5.3 Test Results

Figures 4 to 7 show a table containing the eleven-point interpolated average precision values of all the query results, and the corresponding average P-R curve, for each of the considered websites.

Let us start by analyzing the *http://www.cobat.it/* results (Figure 4). As expected, *Google* results fall between our *Base* and *Stem* setups, because Google programmatically establishes whether to use stemming or not on a document by document basis: in this instance, unconditional stemming was more effective. Synonyms and related terms didn't have a big effect on their own, especially for *Q1* and *Q2*, because they already yielded very good results without, so the additional records retrieved actually lowered precision and thus decreased the score for the second query.

Query	Our results							Google
	Base	Stem	Stem + SynRels	Stem + Prof	Stem + Prof + SynRels	Stem + HetProf	Stem + HetProf + SynRels	
Q1	0,04849	0,02857	0,29184	0,54697	0,60435	0,53496	0,59215	0,27273
Q2	0,08089	0,21579	0,26412	0,62044	0,71850	0,61004	0,76318	0,18182
Q3	0,00000	0,00000	0,13943	0,00000	0,46807	0,00000	0,42199	0,00000
Q4	0,00000	0,39532	0,33033	0,67354	0,49408	0,66602	0,49147	0,00000
Q5	0,00000	0,00000	0,04206	0,00000	0,52799	0,00000	0,52095	0,00000
Average	0,02588	0,12794	0,21355	0,36819	0,56260	0,36221	0,55795	0,09091

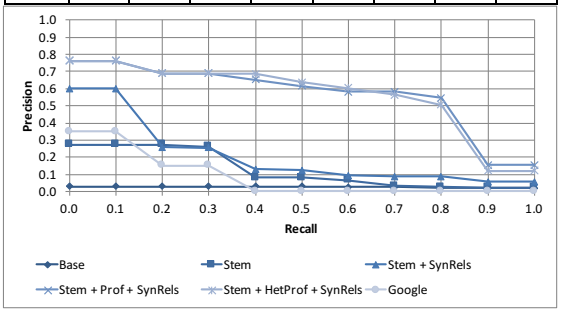

Figure 6: Test results for http://truegoods.com/.

Query	Our results							Google
	Base	Stem	Stem + SynRels	Stem + Prof	Stem + Prof + SynRels	Stem + HetProf	Stem + HetProf + SynRels	
Q1	0,92350	0,91351	0,91351	0,91243	0,90953	0,89566	0,88889	0,46875
Q2	0,00000	0,00000	0,65483	0,00000	0,80483	0,00000	0,64045	0,09869
Q3	0,00000	0,00000	0,00000	0,00000	0,00000	0,00000	0,00000	0,01818
Q4	0,00000	0,00000	0,00817	0,00000	0,02893	0,00000	0,01035	0,00000
Q5	0,00000	0,92603	0,82039	0,92460	0,84677	0,91804	0,80909	0,00000
Average	0,18470	0,36791	0,47938	0,36741	0,51801	0,36274	0,46976	0,11712

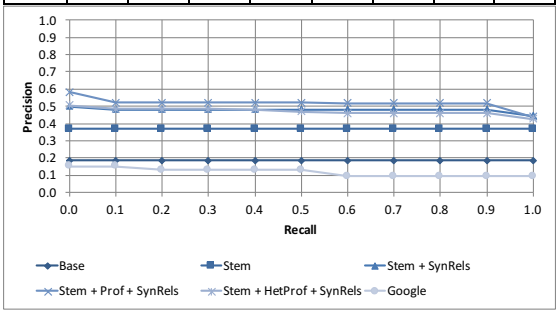

Figure 7: Test results for http://www.gruppozatti.it/.

But on the other hand, they benefited *Prof* and *HetProf* setups greatly, because of the large number of keywords contained within the documents associated to the user profiles: for instance, the use of semantics allowed to match different but very related terms like "battery" and "accumulator", a match that would go unnoticed in a syntactic search.

Going to the second website (Figure 5), *Google* obtained better results than both our *Base* and *Stem* setups, especially because of the greater precision achieved in *Q1* and *Q3*; but the use of synonyms and related terms made up for it with much better results in *Q2*, *Q4* and *Q5*, by exploiting a number of terms correlations such as between "money" and "bookkeeping". Indeed, *Q1* and *Q3* are simple one-word queries that will find matches in any of the relevant documents, while *Q2*, *Q4* and *Q5* are longer and less direct, and thus harder to satisfy for a search engine without additional information in the form of synonyms and related terms. This is also apparent by taking a look at the results of the *Prof* and *HetProf* setups, which greatly improve in their *SynRels* variation.

Looking at the graph in Figure 6, *Google* yielded good results up to the 0,4 recall mark, where the curve plummets to 0,0 precision, meaning most of the relevant documents were not retrieved; this is probably due to indexing issues with this specific website. Our results in this instance are a good example of how computationally determined input, in the form of word stems, synonyms, related terms and profile documents, can turn an apparently impossible query into a manageable one. *Google* and the *Base* setup could not retrieve any records for *Q3*, *Q4* and *Q5*, since they don't include terms that match exactly the ones found in the relevant documents; the *Stem* setup made *Q4* into a succesful query, and the *SynRels* setup did the same for *Q3* and *Q5*, especially in conjunction with *Prof* and *HetProf* (among the exploited terms correlations, the very frequent one between "pet" and "animal").

Our final test (Figure 7) considers a site whose pages contain very little text, thus providing a different task w.r.t. the others. As in most cases, *Google* delivered good results only for *Q1*, the easiest query. Much like the previous websites, a lot of complex queries did not yield satisfactory results for *Google* or *Base*; instead, the *Stem* setup and the *SynRels* setup provide a lot of benefits for *Q5* and *Q2*, while *Q3* and *Q4* were apparently too difficult even with the additional input. Anyway, we see that, on mean, the effect of the semantics and of the user profile is evident from the results even in this specific setting.

6 RELATED WORK

In this section we report some work related to the presented user-aware semantic enterprise search engine. In Figure 8 we propose an analysis of the existing approaches (both academic and commercial), classifying them on the base of two aspects: the user-awareness and the semantics. As mentioned, most approaches do not consider together these aspects and/or not strictly belonging to the enterprise search engine

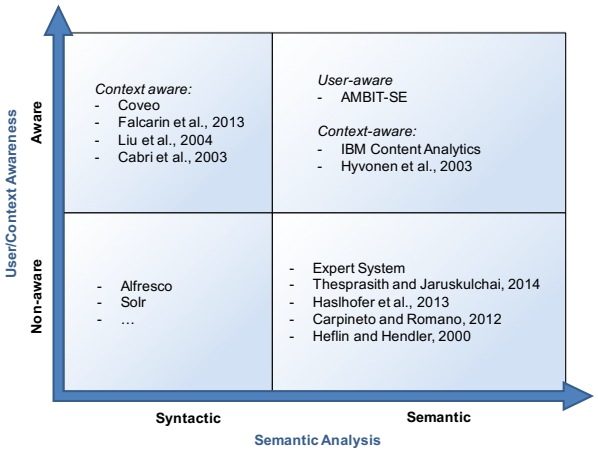

Figure 8: A quadrant for user-aware semantic approaches.

category, so we will discuss related work in three separate subsections: semantic approaches, user-aware approaches and enterprise search engines.

6.1 Semantic Approaches

A broad range of methods for semantic document retrieval has been developed in the context of the Semantic Web, as discussed in (Mangold, 2007), a survey which covers approaches that exploit domain knowledge to process search requests; the authors present a large variety of domain knowledge utilization that comprise automatic query expansion and ontology-driven document retrieval.

The relative ineffectiveness of information retrieval systems is largely caused by the inaccuracy with which a query formed by a few keywords models the actual user information need; one well known method to overcome this limitation is automatic query expansion, whereby the user's original query is augmented by new features with a similar meaning (Carpineto and Romano, 2012). Differently from our approach, complex query expansion techniques such as the ones discussed usually require different parameters to be specified (as also stated in (Abdou and Savoy, 2008)). Generally, there is no single theory capable of finding the most appropriate values (Abdou and Savoy, 2008) and therefore a long process of manual tuning becomes necessary.

An increasing number of document retrieval systems make use of ontologies to help users clarify their information needs and come up with semantic representations of documents. In (Haslhofer et al., 2013), a Simple Knowledge Organization System (SKOS) based term expansion and scoring technique that leverages labels and semantic relationships of SKOS concept definitions is proposed.

Focusing on the necessity of manual intervention, typical semantic retrieval techniques obtain good effectiveness levels only on manually annotated collections and/or with explicit user intervention. In (Thesprasith and Jaruskulchai, 2014), a query expansion technique works on MEDLINE documents which have been manually assigned to controlled MeSH (Medical Subject Headings) vocabularies. The advanced indexing and retrieval method we propose for AMBIT-SE, instead, exploits the semantics of the text while remaining completely automatic.

6.2 User-aware Approaches

Several works in the literature have highlighted the benefits of managing context information and/or proposed techniques and applications exploiting context-awareness capabilities (Bolchini et al., 2011; Liu et al., 2004; Cabri et al., 2003). In particular, a few works are directed towards context modeling, representation, and effective handling. For instance, (Bolchini et al., 2011) proposes to design a context management system which is not application-dependent, while (Villegas and Mller, 2010) reports the result of a study on various context modeling and management approaches. (Liu et al., 2004) proposes a method to derive a user profile based on the search history and on pre-determined category hierarchies. On the other hand, standard search engines such as Google typically provide only very simple IP-address based localization of search results. Most of these approaches, including the ones discussed above in the literature, primarily focus on specific aspects such as external user information or location, do not consider the semantics of the context and/or rely on manual work in order to classify and categorize users and documents.

6.3 Enterprise Search Engines

There is certainly a vast offer of enterprise search engines on the market and in the literature.

The great majority of products does not exhibit a strong focus on ontology-based *semantic* analysis, relying instead on *syntactic* and *hand-coded* rules. Some examples include Alfresco[7], Solr[8].

There are, of course, exceptions to this rule, such as: the SHOE project (Heflin and Hendler, 2000), which requires a domain-ontology where document types correspond to ontology concepts; Expert System's Cogito[9], which provides automated disambiguation, classification, entity extraction, and meta-

[7]http://www.alfresco.com/
[8]http://lucene.apache.org/solr/
[9]http://www.expertsystem.com/it/cogito/

data. However, these systems have no notion of user context. The same can't be said for Coveo[10], a tool specifically oriented to exploit contextual knowledge for dealing with information related to customers and agents. No semantic information, however, is exploited.

On the other hand, there also a small number of systems which exploit, even if in a sometimes limited way, semantic and context information. The Ontogator system (Hyvonen et al., 2003), which is part of an image management and retrieval system, provides an interactive recommendation system which allows the user to browse images based on ontological properties. To exploit user contexts, it introduces views to the ontology that rely on different concept hierarchies, called "facets". Each view represents a specific information-need. IBM's Content Analytics with Enterprise Search[11] exploits a framework called Unstructured Information Management Architecture (UIMA), in order to build analytic applications and to find meanings, relationships and relevant facts hidden in unstructured text. Context information is provided by means of manual annotations. These approaches require manual intervention on the documents and/or adopt a still limited notion of context, i.e. they do not exploit all of the data potentially available on the user, such as the contents of any web page visited, attachment downloaded, etc.

7 CONCLUSIONS

In this paper we have presented AMBIT-SE, a semantic enterprise search engine that takes advantage of user-awareness. To this purpose, the engine exploits textual information (coming from several sources) about the user, and builds a User semantic glossary, which is exploited to enable effective user-aware searches on the retrievable information, stored in the Website semantic glossary. We have tested it with different real websites; the results show that our combined exploitation of synonyms, related terms and user information leads to very good performance, much better than standard syntactic (enterprise) search engines.

With regard to future work, we aim at further optimizing the employed similarity metrics and testing our approach with a wider range of websites. Indeed, the reported experiments consider a good number of cases with different features, but more tests can be useful to further confirm the validity of our approach.

[10]http://www.coveo.com/

[11]https://www.ibm.com/

ACKNOWLEDGEMENTS

This work was supported by the project "Algorithms and Models for Building context-dependent Information delivery Tools" (AMBIT) co-funded by Fondazione Cassa di Risparmio di Modena (SIME 2013.0660).

REFERENCES

Abdou, S. and Savoy, J. (2008). Searching in medline: Query expansion and manual indexing evaluation. *Inf. Process. Manage.*, 44(2):781–789.

Baeza-Yates, R. A. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Beneventano, D., Bergamaschi, S., and Martoglia, R. (2015). Exploiting semantics for searching agricultural bibliographic data. *Journal of Information Science*.

Bergamaschi, S., Martoglia, R., and Sorrentino, S. (2015). Exploiting semantics for filtering and searching knowledge in a software development context. *Knowledge and Information Systems*, 45(2):295–318.

Bernardi, R. (2011). Digital libraries: Ranked evaluation.

Bolchini, C., Orsi, G., Quintarelli, E., Schreiber, F. A., and Tanca, L. (2011). Context modeling and context awareness: steps forward in the context-addict project. *Bulletin of the Technical Committee on Data Engineering*, 34:47–54.

Cabri, G., Leonardi, L., Mamei, M., and Zambonelli, F. (2003). Location-dependent Services for Mobile Users. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems And Humans*, 33(6):667–681.

Carpineto, C. and Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Comput. Surv.*, 44(1):1:1–1:50.

Haslhofer, B., Martins, F., and Magalhães, J. a. (2013). Using skos vocabularies for improving web search. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13 Companion, pages 1253–1258.

Heflin, J. and Hendler, J. (2000). Searching the web with shoe. In *Artificial Intelligence for Web Search. Papers from the AAAI Workshop*.

Hyvonen, E., Saarela, S., and Viljanen, K. (2003). Ontogator: combining view- and ontology-based search with semantic browsing. In *Proceedings of XML Finland*.

Leacock, C. and Chodorow, M. (1998). Combining local context and wordnet similarity for word sense identification. In *WordNet: An electronic lexical database*.

Liu, F., Yu, C., and Meng, W. (2004). Personalized web search for improving retrieval effectiveness. *IEEE Trans. on Knowl. and Data Eng.*, 16(1):28–40.

Mangold, C. (2007). A survey and classification of semantic search approaches. In *Semantics and Ontology*.

Martoglia, R. (2011). Facilitate IT-Providing SMEs in Software Development: a Semantic Helper for Filtering and Searching Knowledge. In *SEKE*, pages 130–136.

Martoglia, R. (2015). Ambit: Semantic engine foundations for knowledge management in context-dependent applications. In *SEKE*, pages 146–151.

Thesprasith, O. and Jaruskulchai, C. (2014). Query expansion using medical subject headings terms in the biomedical documents. In *Intelligent Information and Database Systems - 6th Asian Conference, ACIIDS 2014, Bangkok, Thailand, April 7-9, 2014, Proceedings, Part I*, pages 93–102.

Villegas, N. M. and Mller, H. A. (2010). Managing dynamic context to optimize smart interactions and services. In Chignell, M., Cordy, J., Ng, J., and Yesha, Y., editors, *The Smart Internet*, volume 6400 of *Lecture Notes in Computer Science*, pages 289–318. Springer Berlin Heidelberg.

Xiang, B., Jiang, D., Pei, J., Sun, X., Chen, E., and Li, H. (2010). Context-aware ranking in web search. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, pages 451–458.