A NEW TECHNIQUE FOR IDENTIFICATION OF RELEVANT WEB PAGES IN INFORMATIONAL QUERIES RESULTS

Fabio Clarizia, Luca Greco and Paolo Napoletano
Department of Information Engineering and Electrical Engineering, University of Salerno
Via Ponte Don Melillo 1, 84084 Fisciano, Italy

Keywords: Web search engine, Ontology, Topic model.

Abstract: In this paper we present a new technique for retrieving relevant web pages in informational queries results. The proposed technique, based on a probabilistic model of language, is embedded in a traditional web search engine. The relevance of a Web page has been obtained through the judgment of human beings which, referring to continue scale, have assigned a degree of importance to each of the analyzed websites. In order to validate the proposed method a comparison with a classic engine is presented showing comparison based on a measure of Precision and Recall and on a measure of distance with respect to the measure of significance obtained by humans.

1 INTRODUCTION

Modern Web search engines rely on keyword matching and link structure (cfr. Google and its Page Rank algorithm (Brin, 1998)), but the semantic gap is still not bridged. Indeed, semantics of a web page is defined by its content and context; understanding of textual documents is still beyond the capability of today's artificial intelligence techniques and many multimedia features of a web page make the extraction and representation of its semantics more difficult. But the most critical aspect regards the intrinsic ambiguity of language, which makes the task far harder: when performing informational queries, existing web search engines often show results not close enough to user intentions. As well known any writing process can be thought as a process of communication where the main actor, namely the writer, encodes his intentions through the language. Therefore the language can be considered as a code that conveys what we can call "meaning" to the reader that performs a process for decoding it. Unfortunately, due to the accidental imperfections of human languages, contingent imperfections may occur then both encoding and decoding processes are corrupted by "noise", hence are in practice ambiguous. Nevertheless Semantic Web (Berners-Lee et al., 2001) and Knowledge Engineering communities are both confronted with the endeavor to design different tools and languages for describing semantics in order to avoid the ambiguity of the encoding/decoding process. In the light of this discussions specific language has been introduced, RDF (Resource Description Framework), OWL (Ontology Web Language), etc., to support the creator (writer) of documents in describing semantic relations between concept/words, namely by adding description on the document's data: the *metadata*. During such a process of creation all the elements of ambiguity should be avoided because of use of a shared knowledge represented through the concept of *ontology*. Actual web pages/resources are coded through HTML and/or XHTML languages which, even permitting minimal data descriptions, aren't appropriate, from a technological point of view, to provide purely semantic description. As a consequence the Web should be entirely re-written in order to semantically arrange the content of each web pages, but this process cannot be realized yet, due to the huge amount of existent data and absence of definitive tools for managing and manipulating those new languages. In the meantime, waiting for the semantic web starting, we could design tools for automatically revealing and managing semantics of the previous web by using methods and tools that don't ground on any Web Semantic specification. In this direction several efforts for providing models of language has been made by the Natural Language Processing community, principally developed in a probabilistic framework (Manning and Schütze, 1999). Actually also the Information Retrieval community has focused its atten-

Clarizia F., Greco L. and Napoletano P. (2010).
A NEW TECHNIQUE FOR IDENTIFICATION OF RELEVANT WEB PAGES IN INFORMATIONAL QUERIES RESULTS.
In *Proceedings of the 12th International Conference on Enterprise Information Systems - Information Systems Analysis and Specification*, pages 70-79
DOI: 10.5220/0002903100700079

tion on probabilistic models of language, for instance methods like the probabilistic Latent Semantic Index (pLSI) (Hofmann, 1999) first and the Latent Dirichlet Allocation (LDA) (Blei et al., 2003) later, are well know probabilistic methods for automatic categorization of documents.

As mentioned above the existing Web search engines primarily solve syntactic query, and as a side effect the majority of search results do not completely satisfy user intentions, especially when the queries are informational. This work will show as a classic search engine can improve its search results, and then bring them closer to the user intentions, using a tool for automatic creation and manipulation of ontologies based on an extension of LDA, quoted above, called the *topic model*. To this end, a new search engine, *iSoS*, based on the existing open source software Lucene, was developed. More details will be explained in the next sections together with experiments aimed to make a comparison between iSoS and a classic engine behaviours . The comparing method relies on an innovative human judgment based procedure, which we broadly discuss next, and which represents the real core of this paper.

2 *iSoS*: A TRADITIONAL WEB SEARCH ENGINE WITH ADVANCED FUNCTIONALITIES

As discussed above, iSoS is a web search engine with advanced functionalities; it's a web based server-side application, entirely written in Java and Java Server Pages programming languages, which embeds a customized version of the open source API Apache Lucene [1] for indexing and searching functionalities. In next sections we show the main properties and its functionalities of iSoS framework. Some use cases will be shown, including how to build a new index, include one or more ontologies, perform a query, build a new ontology.

2.1 Web Crawling

Each web search engine works by storing information about web pages retrieved by a Web crawler, a program which essentially follows every link it finds browsing the Web. Due to hardware limitations, our application doesn't implement its own crawling system, but a smaller environment is created in order to evaluate performance: the crawling stage is performed by submitting a specific query to the famous web search engine Google (www.google.com), and extracting the URLs from the retrieved results. Then the application downloads the corresponding web pages to be collected in specific folders and indexed. The GUI allows users to choose the query and the number of pages they want to index.

[1] http://lucene.apache.org/

2.2 Indexing

The main aim of the indexing stage is to store statistics about terms to make their search more efficient. A preliminary document analysis is needed in order to recognize tag, metadata, informative contents: this step is often referred as *parsing*. A standard Lucene index is made of a document sequence where each document, represented as an integer number, is a *field* sequence, with every field containing *index terms*; such an index belongs to the *inverted index* family because it can list, for a term, the documents that contain it. Correct parsing helps to make well categorized field sets, improving subsequent searching and scoring. There are different approaches to web pages indexing. For example, it must be said that some engines don't index whole words but only their stems. The stemming process reduces inflected words to their root form and is a very common element in query systems such as Web search engines, since words with the same root are supposed to bring similiar informative content. In order to avoid indexing of common words such as prepositions, conjuctions, articles which don't bring any additional information, *stopwords* filtering can also be adopted. Since stemmed words indexing and stopwords filtering often result in a lack of search precision, although they could help reducing the index size, they're not the choice of important search engines (like Google). For this application, we developed a custom Lucene analyzer which allows to index both words and their stems without stopwords filtering; it is possible than to include in the searching process ontologies made of stemmed words and thus optimize ontology-based search without penalizing original query precision.

2.3 Searching and Scoring

The earth of a search engine lays in its ability to rank-order the documents matching a query. This could be done through specific score computations and ranking policies. Several information retrieval (IR) operations (including scoring documents on a query, documents classification and clustering) often rely on the *vector space model* where documents are represented as vectors in a common vector space (Christopher D. Man-

Table 1: An example of ontology for the topic *Apple*.

Word 1	Word 2	Relation factor
fruit	popular	0.539597
fruit	juic	0.530942
fruit	mani	0.539458
fruit	seed	0.531137
orchard	popular	0.530548
cider	popular	0.531391
mani	tree	0.535011
mani	fruit	0.539458

ning and Schtze, 2008). For every document d, a vector $\vec{V(d)}$ can be considered, with a component for each dictionary term computed through *tf-idf* weighting. The *tf-idf* weighting assings to term t a weight in a document d given by $tf\text{-}idf_{t,d} = tf_{t,d} \times idf_t$, where $tf_{t,d}$ is the term frequency and idf_t is the *inverse document frequency* defined as $idf_t = \log \frac{N}{df_t}$ with N being the total number of documents and df_t being the number of documents containing the term t. In this model, a query can be also represented as a vector $V(q)$, allowing to score a document d by computing the following *cosine similarity*:

$$score(q,d) = \frac{\vec{V(q)} \cdot \vec{V(d)}}{|\vec{V(q)}||\vec{V(d)}|}$$

where the denominator is the product of the *Euclidean lengths* and compensates the effect of document length. Lucene embeds very efficient searching and scoring algorithms based on this model and on the Boolean model. In order to perform ontology-based search, we customized the querying mechanism to fit our needs. Since our ontologies are represented as couples of related words where relationships strenght is described by a real value (Relation factor), we used Lucene Boolean model and term boosting faculties to extend the base query with ontology contributions.

Table 1 shows an example of ontology representation for the topic *Apple*; including such an ontology in the searching process would basically result in performing:

```
((fruit AND popular)^0.539597) OR
((fruit AND juic)^0.530942)...
```

that means to search the couple of words *fruit* AND *popular* with a boost factor of 0.539597 OR the couple of words *fruit* AND *juic* with a boost factor of 0.530942 and so on. The default boost factor is 1 and is used for the original user query. As discussed before, the way documents are indexed allows to perform queries with full or stemmed words; ontologies generated by the ontology builder tool, which will be discussed in the next section, are always made of stemmed words to organize information in a compact fashion avoiding redundancy.

2.4 Ontology Builder

The *Ontology builder* is an automatic tool for construction of ontology based on the extension of the probabilistic topic model introduced in (T. L. Griffiths, 2007) and (Blei et al., 2003). This method has been deeply illustrated in (Colace et al., 2008), next we will show the main idea behind it. The original theory mainly asserts a semantic representation in which word meanings are represented in terms of a set of probabilistic topics z_i where the statistically independence among words w_i and the "bags of words" assumptions were made. The "bags of words" assumption claims that a document can be considered as a feature vector where each element in the vector indicates the presence (or absence) of a word, where information on the position of that word within the document is completely lost. This model is generative and it allows to solve several problems, including the word association problem, that is a fundamental for the automatic ontology building method. Such a problem was studied for demonstrating what is the role that the associative semantic structure of words plays in episodic memory. In the topic model, word association can be thought of as a problem of prediction. Given that a cue if presented, what new words might occur next in that context? By analyzing those associations we can infer semantic relations among words, moreover by applying this method for automatic interpretation of a document, we can infer all the semantic relations among words contained in that document, as a result we could have a new representation of that document: what we call ontology.

Assume we will write $P(z)$ for the distribution over topics z in particular document and $P(w|z)$ for the probability distribution over word w given topic z. Each word w_i in a document (where the index refers to ith word token) is generated by first sampling a topic from the topic distribution, then choosing a word from the topic-word distribution. We write $P(z_i = j)$ as the probability that the jth topic was sampled for the ith word token, that indicates which topics are important for a particular document. More, we write $P(w_i|z_i = j)$ as the probability of word w_i under topic j, that indicates which words are important for which topic. The model specifies the following distribution over words within a document, $P(w_i) = \sum_{k=1}^{T} P(w_i|z_i = k)P(z_i = k)$, where T is the number of Topics.

In through the *topic model* we can build consistent relations between words measuring their degree

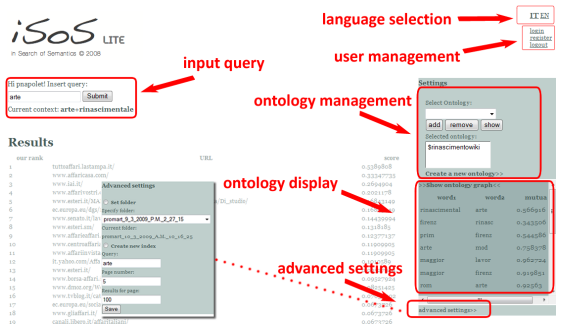

Figure 1: iSoS GUI screenshot.

of dependence, formally by computing joint probability between words, $P(w_i, w_j) = \sum_{k=1}^{T} P(w_i|z_i = k)P(w_j|z_j = k)$.

In this model, the multinomial distribution representing the gist is drawn from a Dirichlet distribution, a standard probability distribution over multinomials. The results of LDA algorithm (Blei et al., 2003) , obtained by running Gibbs sampling, are two matrix:

1. the words-topics matrix Φ: it contains the probability that word w is assigned to topic j;
2. the topics-documents matrix Θ: contains the probability that a topic j is assigned to some word token within a document.

By comparing joint probability with probability of each random variable we can establishes how much two variables (words) are statistically dependent, in facts the hardness of such statistical dependence increases as mutual information measure increases, namely, $\rho = \log|P(w_i, w_j) - P(w_i)P(w_j)|$, where $\rho \in [0, -1]$, after a normalization procedure. By selecting hard connections among existing all, for instance choosing a threshold for the mutual information measure, a graph for the words can be delivered. As a consequence, an ontology can be considered as set of pair of words each of them having its mutual informational value, see Table 1.

3 iSoS IN PRACTICE

Figure 1 shows a screenshot of iSoS home page. The GUI looks very simple, with few essential elements: on the left side at the top there's the main logo and below the input query section, where the searching context is also shown. The searching context refers to the set of documents selected during the crawling stage for a specific generator query. On the right side at the top there are the language selection and the user management modules. This version allows to choose only two languages: English and Italian; different stemming algorithms and stopword sets are used depending on the chosen language. User registration can be done through a specific form and is required to create, store and recall user ontologies created with the ontology builder tool; the lateral panel is organized in three sections:

- *Ontology management*. This section allows to select one or more ontologies from the global set or user customized set to add or remove them from the searching process by clicking on *add* or *remove* buttons respectively.The option *Create a new ontology* allows registered users to launch the *Ontology Builder* tool, which will be discussed in the next subsection.
- *Ontology display*. The *show* button allows to display ontologies in their native tabular form. It is also possible to show their graphic representation by clicking on *Show ontology graph*.
- *Advanced settings*. By clicking on *advanced settings*, other fields are displayed: *set folder* option allows to change the searching context with another available; *create new index* option allows to create a new index by specifying the generator query, the number of Google search pages, the number of results per page. The default context is the last generated one.

Results, shown in the central part of the screen, include URL links and Lucene scores.

3.1 Ontology Builder in Practice

By using this method for building ontology we are able to catch the main topics treated in a document and provide a unique structure for each document, for more details on the main properties of this method see (Colace et al., 2008). A user could declare his intentions by writing a paper, even few lines, and thanks to this automatic method we can extract the core of his intentions. Alternatively one could build an ontology providing a text that discusses the topics of which he is interested in. At this point, whatever was the origin of this ontology, you can use the same to resolve informational queries of a given context. We demonstrate below that the use of this technique significantly improves the quality, in terms of relevance, of the results obtained by our search engine.

This tool allows registered users to create their customized ontologies by simply uploading a document set they choose to describe a given topic. Figure 2 shows *Ontology builder* GUI; it looks very similiar to iSoS main page, with a lateral panel organized in three sections:

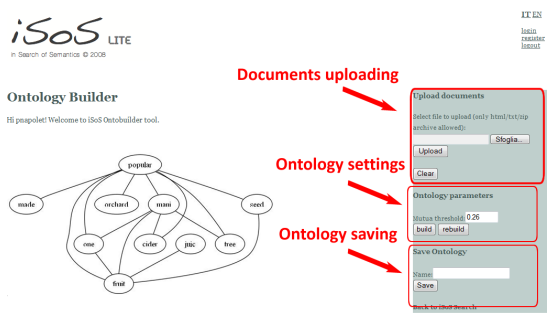

Figure 2: iSoS Ontology builder screenshot.

- *Documents uploading*. It's possible for users to import single documents (txt, HTML pages) or a zip archive containing the document set.
- *Ontology settings*. The user can choose a threshold for the relation factor and build the corresponding ontology by clicking on the *build* button. Once the ontology has been built for the first time, the user can vary the number of nodes by selecting a different threshold and by clicking on the *rebuild* button.
- *Ontology saving*. The user can save the customized ontology by choosing a name and clicking on the *Save* button. Stored ontologies are then available in the ontology management section to be included in the searching process.

4 PERFORMANCE EVALUATION

In order to evaluate search engines performance, different measures can be taken into account: query response time, database coverage, index freshness and availability of the web pages as time passes (Bar-Ilan, 2004), user effort, retrieval effectiveness. Since the main aim of this paper is to point out improvements on retrieval effectiveness due to the introduction of ontologies into the searching process, we will only concentrate on aspects regarding the relevancy of top documents retrieved. When testing a particular search engine, it can be useful to make comparisons with behaviours coming from different engines in order to put in evidence the strenghts and weaknesses of the system under test. In this study, a first evaluation stage has been conducted by comparing iSoS behaviour with a Google Custom Search Engine (CSE); the reasons for using Google CSE instead of Google standard deal with iSoS limited crawling faculties: we had to ensure that both engines performed searches on the same corpus of web pages and Google CSE allows to specify an URLs list of the Web pages to be considered in the searching process. To evaluate retrieval effectiveness, the most commonly used criteria are precision and recall (Christopher D. Manning and Schtze, 2008): the former is defined as the fraction of retrieved documents that are relevant, the latter is the fraction of relevant documents that are retrieved. The huge and ever-changing domain of Web systems makes impossible to calculate true recall, which would require the knowledge of the total number of relevant items in the collection. So recall calculation is often approximated or eventually omitted (Heting Chu, 1996) when comparing search engines performances. In this paper we use a modified recall evaluation introduced by Vaughan (Vaughan, 2004) in a previous work which relies on human judgement. Unfortunately, as well known, the precision and recall measure doesn't take into account the subjective relevance of the retrieved documents. Indeed previous studies have emphasized that human subjects can make relevance judgements on a continous scale(Howard Greisdorf, 2001), so we found useful to get through a second evaluation stage relying on continous human ranking which can be seen as the ideal ranking reference and provides a better term of comparison between the systems being evaluated. In the following sections we describe how the experimental stage was carried out.

4.1 Topics and Search Queries Selection

When human relevance judgement is involved, a large variety of factors can bias the results as the concept of relevance is very subjective. Previous studies have emphasized that relevance judgements can only be made by people who have the original information needs (Michael Gordon, 1999), so topics and search queries selection should involve people who make the judgement. Since fifty people have been involved in the judging task, five group of ten people have been formed and each group has defined a common information need in Italian. As a result, five topics and related queries were designed:

1. Topic: *Arte Rinascimentale*[2].
   Query: *Arte Rinascimentale*.
   Query referred to later in this paper as AR.
2. Topic: *Evoluzione della lingua italiana*[3].
   Query: *Evoluzione della lingua italiana*.
   Query referred to later in this paper as ELI.
3. Topic: *Storia del teatro napoletano*[4].

[2]Italian translation for *Renaissance art*
[3]Italian translation for *Evolution of italian language*
[4]Italian translation for *History of Neapolitan theatre*

Query: *Storia del teatro napoletano*.
Query referred to later in this paper as STN.

4. Topic: *Storia dell'opera italiana*[5].
Query: *Storia dell'opera italiana*.
Query referred to later in this paper as OPI.
5. Topic: *Origini della mozzarella* [6].
Query: *Origini della mozzarella di bufala*.
Query referred to later in this paper as OMB.

For each topic, three hundred pages have been downloaded from the Web to be indexed by iSoS search engine and their URLs have been used to program a Google Custom Search Engine, allowing both engines to perform searches on the same corpora of documents.

4.2 Ontology Building and Web Pages Retrieved

In order to feed ontology builder and produce ontologies for experiments, we asked five people with great skill or knowledge about chosen topics to provide a set of documents better describing information needs taken into account. Once ontologies have been built, we have performed each query on both iSoS and Google CSE. To better evaluate the ontology contribution, we have also performed each query on iSoS without including any ontology but with a query extension: we simply added ontology terms to the query. Then, for each query we obtained 30 pages, corresponding to the top 10 pages retrieved by each engine, which were merged in order to make the set of pages to be ranked by human subjects for that particular topic. As a result af the merge, the number of pages in the sets AR, OPI, ELI, STN, OMB were 17, 19, 20, 19, 17 respectively.

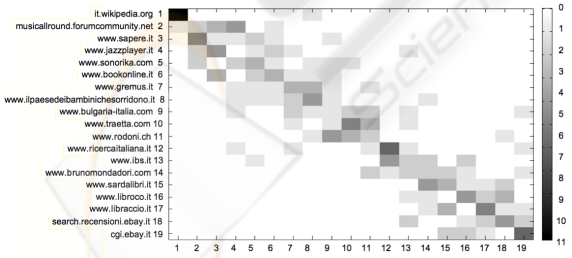

Figure 3: OPI Borda Count results before meeting.

4.3 Human Ranking of the Web Pages

Subjects in the study were graduates in various disciplines ranging from engineering (information, electronic, management) to the literature and economics disciplines that were involved in a university training course. People were divided into five group of ten people each. Each group evaluated a set of documents related to its own query. One must consider that the need behind a web search could be not only informational but navigational (searching for the url of a specific site) or transactional (searching for e-commerce sites...).

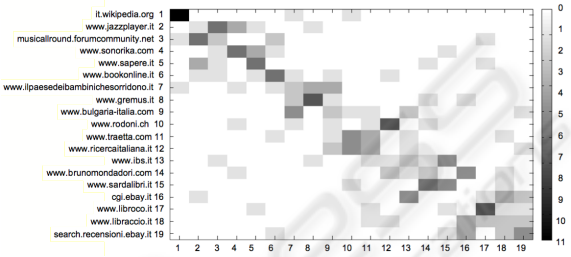

Figure 4: OPI Borda Count results after meeting.

Due to the kind of ontologies produced, when deciding on relevancy the subjects were asked to judge positively only informational Web pages without following any link on the page. Each subject had to rank the pages according to his/her criteria and write down those criteria. After that, each group met discussing about their ranking and criteria in order to improve ranking quality and reduce the effects of unusual ranking by individual subjects. Finally, they were allowed to adjust personal ranking if necessary. In order to derive an average human behaviour, the Borda Count method (Saari, 2001) was applied to both before and after meeting results. Figure 3 shows OPI human ranked results before the meeting together with a graphical representation of Borda count method outcomes. Such a representation provides a simple way of investigating subjects degree of agreement for each result with black being the maximum (all the eleven subjects agreed on the ranking) and white being the minimum (nobody voted for that position). Comparing the results with the ones shown in Figure 4, which refers to the after meeting case, more darker boxes laying closer to the diagonal line of the graph can be seen, showing a better agreement on the final ranking as expected. Therefore, for all the queries we chose as human being behaviour the results coming from the Borda Count method applied to the after meeting case. We decided to limit attention to the top 10 pages because typically users visit only top pages retrieved (Silverstein et al., 1999). Tables 2,3, 4,5, 6 show top 10 human ranked results for each query highlighting iSoS and Google CSE positioning for each page. Complete URLs of pages are not reported due limitations in space.

[5]Italian translation for *History of italian opera*
[6]Italian translation for *Origins of buffalo mozzarella*

Table 2: Results obtained for the query AR.

	Human ranked URLs	iSoS	Google
1	www.artistiinrete.it	3	5
2	www.bilanciozero.net	2	> 10
3	it.encarta.msn.com	5	3
4	www.firenze-online.com	1	> 10
5	it.wikipedia.org	4	1
6	www.arte.go.it	8	> 10
7	digilander.libero.it	9	> 10
8	www.visibilmente.it	6	7
9	www.salviani.it	> 10	4
10	www.arte-argomenti.org	7	> 10

Table 3: Results obtained for the query ELI.

	Human ranked URLs	iSoS	Google
1	blogs.dotnethell.it	3	> 10
2	it.wikipedia.org	1	1
3	www.letteratour.it	2	> 10
4	www.nonsoloscuola.net	4	> 10
5	digilander.libero.it	7	> 10
6	xoomer.virgilio.it	6	> 10
7	www.etx.it	8	> 10
8	www.regione.emilia-romagna.it	> 10	10
9	www.tesionline.com	> 10	6
10	www.tesionline.it	> 10	3

Table 4: Results obtained for the query OPI.

	Human ranked URLs	iSoS	Google
1	it.wikipedia.org	4	10
2	www.jazzplayer.it	2	> 10
3	musicallround.forumcommunity.net	3	> 10
4	www.sonorika.com	5	> 10
5	www.sapere.it	6	> 10
6	www.bookonline.it	1	> 10
7	www.ilpaesedeibambinichesorridono.it	10	> 10
8	www.gremus.it	> 10	1
9	www.bulgaria-italia.com	8	> 10
10	www.rodoni.ch	9	> 10

Table 5: Results obtained for the query STN.

	Human ranked URLs	iSoS	Google
1	www.sottoilvesuvio.it	1	7
2	www.blackwikipedia.org	3	> 10
3	it.wikipedia.org/wiki	2	1
4	xoomer.virgilio.it	5	5
5	www.webalice.it	6	> 10
6	www.laboriosi.it	8	> 10
7	www.denaro.it	> 10	4
8	www.gttempo.it	7	> 10
9	www.teatroantico.org	10	> 10
10	azzurrocomenapoli.myblog.it	4	> 10

4.4 Precision and Recall Evaluation

Precision is a measurement always present in formal information retrieval problems. As described before, it is defined as the fraction of retrieved documents that are relevant so it depends on how relevancy judgements were made. Binary relevance judgements are often used, also in TREC experiments: a document is relevant to the topic or it's not (Voorhees, 2003). Several studies have used multi-level rather then binary relevance judgements, but all these kinds of discrete relevance scores suffer from the possibility of giving the same score to different documents. So they are not very useful when evaluating ranking results. However, in this study we use a variant of standard precision-recall evaluation based on human ranked results; in particular, we consider the top 10 human ranked results as relevant and this assumption allows us not to care about the real amount of relevant documents in the corpus, that would be very hard to calculate in a precise manner. One must consider that our target is to compare iSoS and Google CSE behaviours with the human one, so this method appears to be very effective although approximated. For each set of results, precision and recall values have been plotted to give a *precision-recall interpolated curve*: the interpolated precision p_{interp} at a certain recall level r is defined as the highest precision found for any recall level $r' \geq r$, $p_{interp}(r) = \max_{r' \geq r} p(r')$.

The traditional way of representing is the 11-point interpolated average precision . For each information need, the interpolated precision is measured at the 11 recall levels of 0.0, 0.1, 0.2, ..., 1.0. For each recall level, we then calculate the arithmetic mean of the interpolated precision at that recall level for each information need in the corpus. A composite precision-recall curve showing 11 points can then be graphed. Figures 5, 6, 7, 8, 9 show precision-recall interpolated graphs for the queries AR, ELI, OPI, OMB, STN respectively. It can be seen that for all the examinated cases iSoS displays a better performance than Google CSE, with OPI being the best case. However, as said before, precision and recall measurements rely on binary relevancy judgements so we don't find them accurate enough to make a good comparison. In the next section, we propose an alternative method to compare iSoS and Google CSE behaviours with the human one.

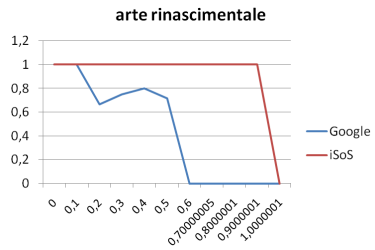

Figure 5: Precision - Recall for AR query.

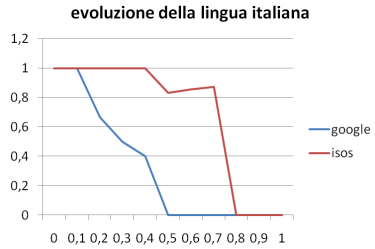

Figure 6: Precision - Recall for ELI query.

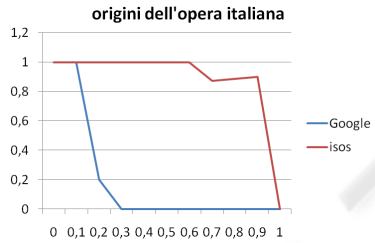

Figure 7: Precision - Recall for OPI query.

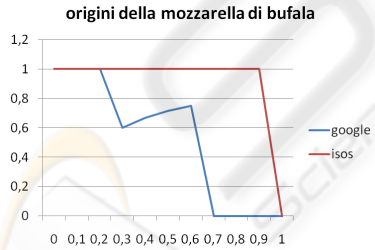

Figure 8: Precision - Recall for OMB query.

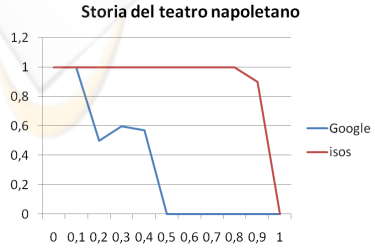

Figure 9: Precision - Recall for STN query.

Table 6: Results obtained for the query OMB.

	Human ranked URLs	iSoS	Google
1	magazine.paginemediche.it	6	> 10
2	www.caseificioesposito.it	1	> 10
3	www.agricultura.it	2	> 10
4	it.wikipedia.org	3	2
5	www.mozzarelladibufala.org	> 10	1
6	www.ciboviaggiando.it	9	6
7	www.sito.regione.campania.it	8	5
8	www.aversalenostreradici.com	4	> 10
9	www.tenutadoria.it	7	7
10	www.bortonevivai.it	5	8

4.5 Relevance Evaluation

A better performance evaluation can be done by comparing iSoS and Google CSE results with the human ranking which we consider the reference behaviour. Figure 10 provides a graphical representation of iSoS, Google CSE and human being (HB) behaviours for the query AR; being the reference, HB behaviour is displayed as a blue line while iSoS and Google CSE results are displayed as red and green curves respectively. This representation allows to make a fast visual comparison between the different behaviours: the more a curve lays close to the blue line, the more the search engine represented by that curve exhibits an ideal behaviour. The graph region to be considered spreads to the tenth result, as said before. The numbers next to the points on the blue line represent the *Mean Square Error* evaluated from the subjects preferences for that position. Another useful representation is shown in Figure 11 and displays the distance of the HB ranking from iSoS ranking (red bar) and from Google CSE ranking (green bar). This kind of analysis has been also conducted for the other queries and is showed in figures 12, 13, 14, 15, 16, 17, 18, 19. It can be seen that iSoS exhibits a better average behaviour for all the cases, with OPI being the best one. Although these outcomes seem close to the precision-recall ones, it must be said that this method helps to better highlight also iSoS weaknesses, which can be seen, for example, on the last three results for the query ELI (Figure 13).

5 CONCLUSIONS

In this work we have seen how the current web search engines are not able to resolve in an appropriate way informational queries. We have also shown how the results of a classic search engine can be improved through the use of an innovative search technique based on using a particular ontology. This ontology

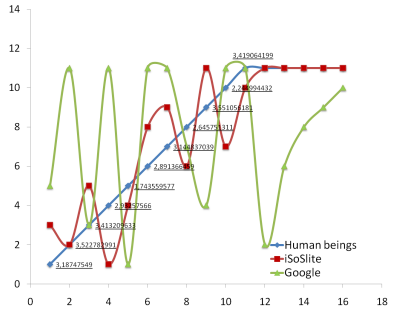

Figure 10: AR comparison.

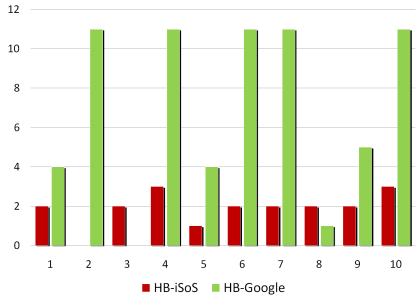

Figure 11: AR comparison.

is obtained by using a technique based on a model of language called probabilistic topic model which is based on the LDA technique, now important in the Information Retrieval community. An ontology of this type is able to capture clearly the main topic of interest, so that when used in a simple informational query task, it can certainly improve the quality of results returned. The proposed technique was validated through a comparison with a classic search engine and a comparison with a measure of significance obtained by experiments with human beings. The experiments were conducted for different contexts and for each of them were asked different groups of human beings to assign judgments of relevance to the set of web pages collected by unifying results obtained either with our search engine and the classic one. The results obtained have confirmed that the proposed technique

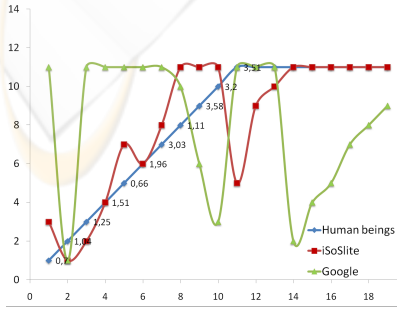

Figure 12: ELI comparison.

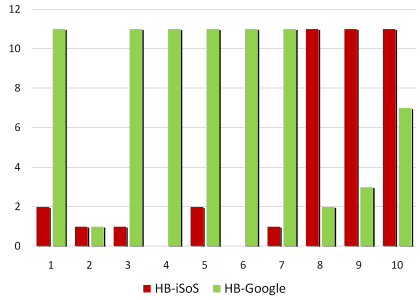

Figure 13: ELI comparison.

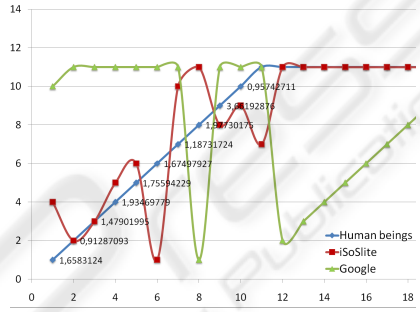

Figure 14: OPI comparison.

certainly increases the benefits in terms of relevance of the results obtained, which in some way therefore respect the user intentions. We have also discussed about the opportunity of developing new techniques for manipulating the language, mainly based on probabilistic models, and consequently use those technique for handle semantic information. This is an important point of our discussion, because actually an enormous quantity of information is on the web, but few are the tools that are able to retrieve in this huge set of information something that really respects users intentions. To this end, the Semantic Web community is working towards the introduction of new technologies and tools to make the enormity of the information on the web handle. Despite this technological paradigm shift has not happened yet and is not at the gates, so in the meantime, tools such as those treated

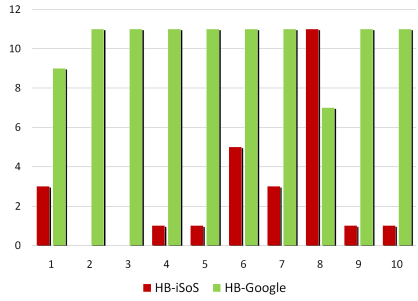

Figure 15: OPI comparison.

in this work can be really useful.

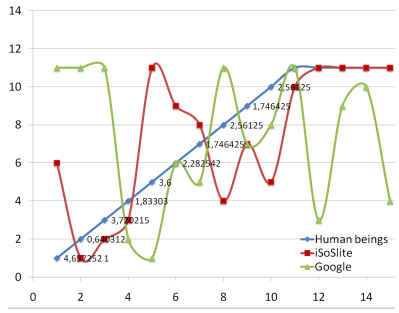

Figure 16: OMB comparison.

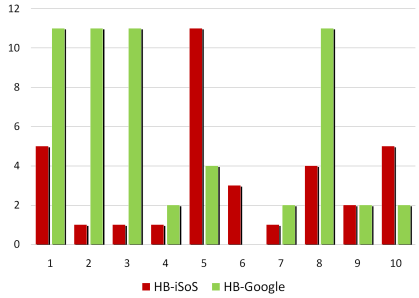

Figure 17: OMB comparison.

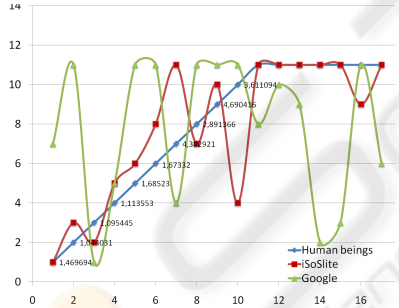

Figure 18: STN comparison.

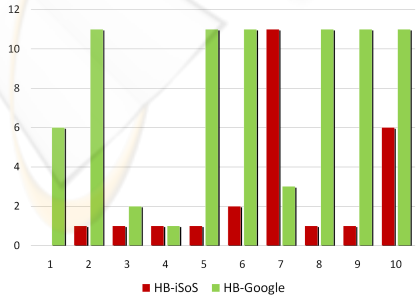

Figure 19: STN comparison.

REFERENCES

Bar-Ilan, J. (2004). Methods for measuring search engine performance over time. *Journal of the American Society for Information Science and Technology*, 53(308–319).

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific American*, May.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(993–1022).

Brin, S. (1998). The anatomy of a large-scale hypertextual web search engine. In *Computer Networks and ISDN Systems*, pages 107–117.

Christopher D. Manning, P. R. and Schtze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Colace, F., Santo, M. D., and Napoletano, P. (2008). A note on methodology for designing ontology management systems. In *AAAI Spring Symposium*.

Heting Chu, M. R. (1996). Search engines for the world wide web: a comparative study and evaluation methodology. In *In Proceedings of the 59th annual meeting of the American Society for Information Science*, pages 127–135.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the Twenty-Second Annual International SIGIR Conference*.

Howard Greisdorf, A. S. (2001). Median measure: an approach to ir systems evaluation. *Information Processing and Management*, 37(6)(843–857).

Manning, C. D. and Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, USA.

Michael Gordon, P. P. (1999). Finding information on the world wide web: the retrieval effectiveness of search engines. *Information Processing and Management*, 35(141–180).

Saari, D. G. (2001). *Chaotic Elections! A Mathematician Looks at Voting*. American Mathematical Society, Providence.

Silverstein, C., Marais, H., Henzinger, M., and Moricz, M. (1999). Analysis of a very large web search engine query log. *ACM SIGIR Forum*, 40(677–691).

T. L. Griffiths, M. Steyvers, J. B. T. (2007). Topics in semantic representation. *Psychological Review*, 114(2):211–244.

Vaughan, L. (2004). New measurements for search engine evaluation. *Information Processing and Management*, 40(677–691).

Voorhees, E. M. (2003). Overview of trec 2003. In *In Proceedings of the 12th Text Retrieval Conference*, pages 1–13.