Semantically Driven Snippet Selection for Supporting Focused Web Searches

IRAKLIS VARLAMIS
Harokopio University of Athens
Department of Informatics and Telematics, 89, Harokopou Street, 176 71, Athens, Greece
varlamis@hua.gr

and

SOFIA STAMOU
Patras University
Computer Engineering and Informatics Department, 26500, Greece
stamou@ceid.upatras.gr

Millions of people access the plentiful web content to locate information that is of interest to them. Searching is the primary web access method for many users. During search, the users visit a web search engine and use an interface to specify a query (typically comprising a few keywords) that best describes their information need. Upon query issuing, the engine's retrieval modules identify a set of potentially relevant pages in the engine's index, and return them to the users, ordered in a way that reflects the pages' relevance to the query keywords. Currently, all major search engines display search results as a ranked list of URLs (pointing to the relevant pages' physical location on the web) accompanied by the returned pages' titles and small text fragments that summarize the context of search keywords. Such text fragments are widely known as snippets and they serve towards offering a glimpse to the returned pages' contents. In general, text snippets, extracted from the retrieved pages, are an indicator of the pages' usefulness to the query intention and they help the users browse search results and decide on the pages to visit. Thus far, the extraction of text snippets from the returned pages' contents relies on statistical methods in order to determine which text fragments contain most of the query keywords. Typically, the first two text nuggets in the page's contents that contain the query keywords are merged together to produce the final snippet that accompanies the page's title and URL in the search results. Unfortunately, statistically-generated snippets are not always representative of the pages' contents and they are not always closely related to the query intention. Such text snippets might mislead web users in visiting pages of little interest or usefulness to them. In this article, we propose a snippet selection technique, which identifies within the contents of the query relevant pages those text fragments that are both highly relevant to the query intention and expressive of the pages' entire contents. The motive for our work is to assist web users make informed decisions before clicking on a page in the list of search results. Towards this goal, we firstly show how to analyze search results in order to decipher the query intention. Then, we process the content of the query matching pages in order to identify text fragments that highly correlate to the query semantics. Finally, we evaluate the query-related text fragments in terms of coherence and expressiveness and pick from every retrieved page the text nugget that highly correlates to the query intention and is also very representative of the page's content. A thorough evaluation over a large number of web pages and queries suggests that the proposed snippet selection technique extracts good quality text snippets with high precision and recall that are superior to existing snippet selection methods. Our study also reveals that the snippets delivered by our method can help web users decide on which results to click. Overall, our study suggests that semantically-driven snippet selection can be used to augment traditional snippet extraction approaches that are mainly dependent upon the statistical properties of words within a text.

---

1. INTRODUCTION

The advent of the web has brought people closer to information than ever before. Web search engines are the most popular tool for finding useful information about a subject of interest. What makes search engines popular is the straightforward and natural way via which people interact with them. In particular, people submit their requests as natural language queries and they receive in response a list of URLs that point to pages, which relate to the information sought.

Retrieved results are ordered in a way that reflects the pages' importance or relevance to a given query. Despite the engines' usability and friendliness, people are oftentimes lost in the information provided to them, simply because the results that they receive in response to some query comprise of long URL lists. To help users locate the desired information, search engines accompany retrieved URLs with snippets of text, which are extracted either from the description meta-tag, or from specific tags inside the text (i.e. title or headings).

A snippet is a set of usually contiguous text, typically in the size of a paragraph, which offers a glimpse to the retrieved page's content. Snippets are extracted from a page in order to help people decide whether the page suits their information interest or not. Depending on their decisions, users might access the pages' contents simply by clicking on an initial set of retrieved URLs or ignore them and proceed with the next bunch of results.

Most up-to-date web snippet selection approaches extract text passages[1] with keyword similarity to the query, using statistical methods. For instance, Google's snippet extraction algorithm [11] uses a sliding window of 15 terms (or 100 characters), over the retrieved document, to generate text fragments in which it looks for query keywords. The two passages that show up first in the text are merged together and produce the final snippet. However, statistically generated snippets are rough indicators of the query terms co-occurring context but, they lack coherence and do not communicate anything about the semantics of the text from which these are extracted. Therefore, they are not of much help to the user, who must decide whether to click on a URL or not.

Evidently, if we could equip search engines with a powerful mechanism that generates self-descriptive and document expressive text snippets, we would save a lot of time for online information seekers. That is, if we provide users with that piece of text

[1] We use the terms snippet and passage interchangeably to denote the selection of small size text from the full content of a document.

from a page that is the most relevant to their search intention and which is also the most representative extract of the page, we may assist them decide the page is of interest to them before they actually click on it..

In this article, we propose the Semantic Snippet Selection (SemSS) technique, which firstly identifies the query senses and then it looks in the query retrieved pages for text fragments that relate to the query senses. Unlike, existing attempts that select snippets based on the distribution of the query terms in their contents, we suggest the exploitation of semantics in the snippet selection process. Our technique focuses on selecting ***coherent***, ***query-relevant*** and ***expressive*** text fragments, which are delivered to the user and which enable the latter perform focused web searches. At a high level our method proceeds as follows:

- Given a query and set of query matching pages, our method uses a lexical ontology and employs a number of heuristics for disambiguating the query intention.
- Given the disambiguated query and a set of results that relate to the query, our approach identifies within the text of every page, the text fragment that is the most relevant to the semantics of the query.
- Query-relevant text snippets are then evaluated in terms of their lexical elements' coherence, their importance to the semantics of the entire page and their closeness to the query intention.
- Snippets that exhibit the strongest correlation to both the query and the page semantics are presented to the user.

After applying our SemSS technique to a number of searches, we conclude that snippets determined by their semantic correlation to both the query and the document terms yield improved accuracy compared to the snippets that are determined by using only the statistical distribution of query keywords in the pages' extracts. In brief, the contributions of this article are as follows:

- We introduce a measure for quantifying the snippet's closeness to the query intention (usefulness). In our work, a useful snippet is the text fragment in a retrieved page that exhibits the greatest terminological overlap to the query keywords and which is also semantically closest to the query intention.
- We present a metric that estimates the importance and the representation ratio of a snippet with respect to the contents of the entire pages from which this was extracted. Our metric adheres to the semantic cohesion principle and aims at identifying the query focus in the search results.

- We suggest the combination of the above measures, in order to assist web users perform comprehensive and focused web searches in two steps: before and after clicking on the retrieved results. Before clicking on the results, users can view the selected text fragment from the page that best matches their search intentions. Based on the displayed snippet, users may decide whether they want to read the respective page or not. After clicking on a snippet, the users' focus is directly driven to the exact text fragment that contains relevant information to their search intention. In particular, query-relevant text fragments appear highlighted, so that users get instantly the desired information, without the need to go through the entire content of a possibly long document.

The remainder of the article is organized as follows. We begin our discussion with a detailed description of our semantically-driven approach in snippets' selection. Then in Section 3, we experimentally evaluate the effectiveness of our SemSS model in focusing retrieval on the query semantics and we discus obtained results. In Section 4 we review related work and we provide the conclusions of our work in Section 5.

2. MINING QUERY-RELEVANT AND TEXT EXPRESSIVE SNIPPETS

It is common knowledge that web users decide on which results to click based on very little information. Typically, in the web search paradigm, information seekers rely on the retrieved page's title, URL and text snippet that contains their search keywords to infer whether the page is of interest to their search pursuit or not.

Although, the titles given to web pages are manually specified and as such they are generally representative of the pages' contents, nevertheless the text snippets that accompany the pages in the search engine results are automatically extracted and as such they are not always descriptive of the page's thematic content. Given that snippet extraction relies solely on the distribution of the query keywords among their elements, it becomes clear that the selected snippets may communicate incomplete (or even misleading) information about the content of a page and thus they may trigger user clicks on pages that are marginally relevant to the user search pursuits.

Evidently, decisions based on little information are susceptible to be bad decisions. A bad decision is encountered when the user clicks on a result misguided by a text snippet, which is of little relevance to the linked page's contents. In an analogous manner, a bad decision might be when the user decides not to click on a *good* result, simply because the text snippet of the page is poor or seems unrelated to her query intention.

In this section, we present our semantically-driven approach towards the automatic extraction of query-relevant and document-expressive snippets. Our aim is to assist web information seekers make informed decisions, about whether to click on a retrieved result or not.

The basic steps of our approach, as depicted in Figure 1, are:

1. Automatic disambiguation of the query intention based on the semantic analysis of the query relevant pages.
2. Semantic similarity matching between query and text passages, using both terms and implied concepts (candidate passages).
3. Selection of query-similar snippets from the document (useful snippets).
4. Evaluation of the selected snippets' expressiveness to the document contents.
5. Presentation of the query-relevant and document-expressive snippets to the user.

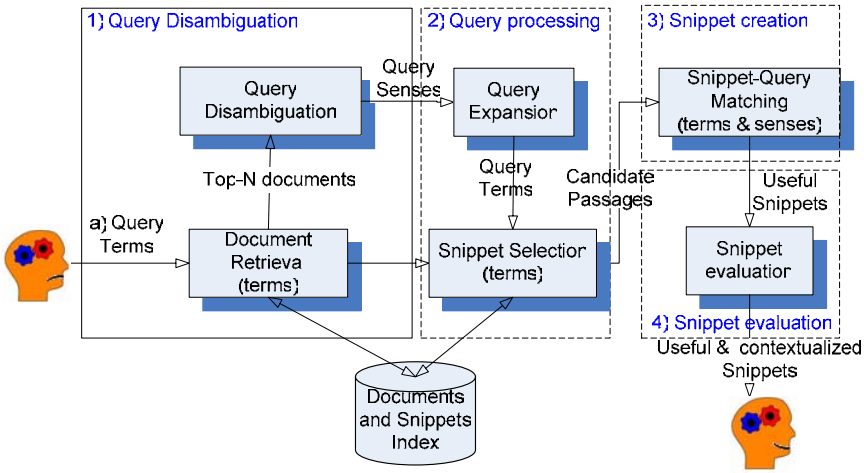

Fig. 1. Snippet Selection Process.

We begin our discussion, with a brief description of our approach towards the identification of the query intention (step 1). Then, in Section 2.2 we describe our semantically-driven approach for extracting candidate text nuggets from a query matching page (step 2). We also discuss how to select the text fragments that are semantically closest to the query intention (step 3). In Section 2.3, we introduce a novel method for evaluating how expressive or else representative is a query-relevant text fragment to the entire content of the page from which it has been extracted (step 4). Finally, in Section 2.4, we discuss how we can put together the derived information in

order to decide on the text nugget that is the most useful to the query intention and also the most expressive of the document's content (step 5).

2.1 Identifying the Query Intention

In order to be able to detect within the contents of a query relevant page the text fragment that is the most relevant to the search query, we firstly need to identify the query intention. This practically, translates into resolving query sense ambiguities. For disambiguating long queries (i.e. containing many terms), we can rely on existing Word Sense Disambiguation (WSD) techniques [41] [42] that resolve the query senses based solely on the query elements. However, numerous studies have shown that the vast majority of web queries are short and under-specified [17]. Short queries are inherently ambiguous, since they lack sufficient contextual data that could specify the query semantics. Therefore, WSD methods have limited applicability towards identifying the correct sense of a short query (Krovetz and Croft [43] [44] report a 2% improve of retrieval performance when manual disambiguation was used). Although, the problem of query sense identification is not new, nevertheless the challenge of deciphering the intention of a query still remains.

The most common WSD approach for resolving the sense of a single term is to select the most frequent sense of the term as the latter is specified in a language thesaurus. In our work, we attempt the automatic identification of the query intention, using WordNet [37] as the semantic resource against which we will look for the appropriate query sense. The reason for focusing on WordNet rather than using some other semantic base, is not only because WordNet is freely available and richly encoded, but also because it has proven to be an excellent resource for deriving the different senses a word has as well as for assisting WSD tasks [18]. Therefore, in our work we rely on WordNet for deriving the possible senses a query might have and for selecting among the candidate query senses the one that is most likely to represent the query intention. For selecting the most suitable query sense among the candidate ones, we explore the contextual elements of the query in the search results of the latter and we resolve the query intention heuristically.

In particular, we rely on the top N (N=20) pages retrieved for a query, we parse them to remove html markup, we apply tokenization, POS-tagging and we eliminate all stop-words from their contents. We then perceive the processed web pages as a small Web corpus of query co-occurrence data, against which we attempt the query sense resolution. Although, this approach adds a computational overhead compared to the straightforward selection of the most frequent query sense as the appropriate one, nevertheless it delivers

more accurate results since it considers all possible query senses and as such it assists the snippet generation process, as we explain next.

Back to the description of the query disambiguation approach, we proceed as follows. We rely on the content terms[2] of the top N query matching pages and we compute their importance weights based on the TF-IDF vector weighting scheme [31]. Formally, the weight $w_{i,j}$ associated with the term t_i in document d_j is given by:

$$w_{i,j} = tf_{i,j} \bullet \log\left(\frac{N}{df_i}\right) \qquad (1)$$

Where $tf_{i,j}$ is the frequency of t_i in d_j, N is the total number of documents considered (in our approach N=20) and df_i is the total number of documents that contain t_i. Having weighted every keyword in the top N retrieved pages, we sort them and we retain the *n* (*n* =25%) most highly weighted terms, as our "disambiguation set". Our next step is to identify the appropriate senses of the pages' keywords. In this respect, we firstly map the pages' highly weighted *n* keywords to their corresponding WordNet nodes and we disambiguate them based on the computations of the their semantic similarity, given by the Wu and Palmer similarity metric [38]. The metric combines the depth of paired concepts in WordNet and the depth of their least common subsumer (LCS), in order to measure how much information the two concepts share in common. The main reason for picking the above similarity metric among the existing ones is because this has been specifically implemented for WordNet-based WSD tasks and because it is simple and restricted on IS-A hierarchical relation types. For an overview of the different similarity metrics employed for word sense resolution see [26]. Moreover, a comparison and an extended evaluation of the most popular WordNet-based measures can be found in [47] and [45]. Recently, we experimentally showed that different relevance measures affect the performance of WSD systems [48]. Based on our finding, we introduced a more sophisticated metric towards WSD that utilizes the full range of WordNet 2.0 semantic relations and also accounts for both the depth and the type of WordNet relations. The evaluation of our metric shows that it has an improved performance compared to many of the existing WSD methods and we are currently in the process of deploying it for query sense resolution tasks.

---

[2] Content terms are nouns, proper nouns, verbs, adjectives and adverbs

However, in the course of the present study we suggest the utilization of the Wu and Palmer similarity metric, since we target our research on dynamic data (i.e. web queries) and hence we need a WSD measure that is simple to implement, it has been extensively tested by numerous researchers and it scales well with large volumes of data.

According to the Wu and Palmer measure, the similarity between two concepts c_i and c_k that are represented in a hierarchy is given by:

$$\text{Similarity}\,(c_i, c_k) = \frac{2 * \text{depth}\,(\text{LCS}\,(c_i, c_k))}{\text{depth}\,(c_i) + \text{depth}\,(c_k)} \qquad (2)$$

Since the appropriate concepts c_i and c_k for a pair of words, namely w_i and w_k, are not known, our measure selects the combination of concepts which maximize the above similarity and consequently annotates every keyword in the page with the most appropriate WordNet sense (i.e. concept) based on:

$$\text{WordSimilarity}\,(w_i, w_k) = \arg\max_{i,k}\,(\text{Similarity}\,(c_i, c_k)) \qquad (3)$$

Having disambiguated the most highly weighted keywords in the query matching pages, we proceed with the disambiguation of the query term(s). To enable that, we map the query terms to their respective WordNet nodes and we apply the Wu and Palmer metric to compute the similarity values between the candidate query senses and the senses associated to the keywords of the query matching pages. We then rely on the average similarity values between the candidate senses of a query term and the senses appended to the "disambiguation-set" DS (top *n* keywords from the matching pages), which is given by:

$$\text{QueryTerm-Keywords Similarity}\,(q, DS) = \frac{1}{n}\sum_{j=1}^{n}\text{Similarity}\,(q_i, k_j) \qquad (4)$$

Where DS is the disambiguation set, q_i is the i^{th} sense of the query term q, k_j is the corresponding sense of keyword k ($k_j \in DS$) and *n* is the total number of keywords considered. Finally, we pick for each query term, the sense that exhibits the maximum QueryTerm-Keywords similarity value as the correct sense for representing the intention of the query. Based on the above process, we annotate every query term with an

appropriate WordNet sense. This information is subsequently utilized by the SemSS mechanism in order to identify which text fragments in the content of the retrieved pages are the most likely to meet the intention of the query.

2.2 Semantic Selection of the Best Snippet for a Query

Having resolved the query sense or else having *learned* the query intention, we now turn our interest to the description of how we can use this *knowledge* for selecting within the contents of a page the text fragment that is semantically closest to the query intention.

The process begins with the expansion of the disambiguated set of query terms with their synonyms in WordNet. Query expansion ensures that text fragments containing terms that are semantically equivalent but superficially distinct to the query terms, are not neglected in the snippet selection process. The snippet selection that follows, finds all the appearances of the original query terms and their synonyms within the retrieved page. Upon identification of query matching items in the page's text, we define a window size of 35 words (see [39]) around the identified query items and we extract all the passages that contain any of the items in the expanded query-set. All the extracted passages are candidate snippets with respect to the considered query.

To identify within a query relevant page those text snippets that better match the query intention, our method combines (i) the ***terminological overlap*** (expressed by the relevance measure) and (ii) the ***semantic correlation*** (expressed by the quality measure) between the query and snippet sets of concepts.

The ***terminological overlap*** between the query and a snippet is, in rough terms, the intersection of the two item-sets (i.e. query and snippet terms); given that all snippets have similar size and that the items in both sets have been mapped to WordNet nodes. The normalized terminological overlap between a query and a passage indicates the ***relevance*** that a passage has to the query intention and it is formally determined by the fraction of the passage terms that have a semantic relation[3] in WordNet to the query sense, as given by:

---

[3] Out of all the WordNet relation types, in our work we employ: direct hypernymy, (co-)hyponymy, meronymy, holonymy and has instance, as indicative of the query-passage terminological relevance.

$$\text{Relevance}(q, p) = \frac{\sum_{j=1}^{k} qr \bullet \text{TFIDF}(t_j, p)}{qs \bullet \sum_{i=1}^{n} \text{TFIDF}(t_i, p)} \quad (5)$$

Where *k* is the number of terms in passage *p* that are directly linked in WordNet to at least one query term via any of the following relation types: *hypernymy*, *hyponymy*, *meronymy*, *holonymy*, *has- instance* or *co-hyponymy*, *n* is the total number of terms in the passage, *qr* is the number of query terms to which the passage term t_j relates (query relevant terms) and *qs* is the number of terms in the query (query size). Finally, TF-IDF(t_x,p) denotes the importance of term t_x in passage *p*. Passages containing terms that relate to the sense of the query keywords are deemed to be query relevant. However, this is not sufficient for judging the quality or the usefulness that the candidate passages have to the query intention.

To ensure that the snippet extracted from a query matching page contains only good quality passages, we semantically correlate the expanded query and the query-relevant passage. To derive correlations, we rely on the semantic similarity between the terms of a query and the terms in each of query relevant passages. Formally, the metric we use for estimating the similarity between a query term q and the passage S is based on equation (4) and given by:

$$\text{QueryTerm-PassageSimilarity}(q, S) = \arg\max_{k} \text{QueryTerm-KeywordsSimilarity}(q, S_k) \quad (6)$$

Where q represents the query term and S_k denotes the disambiguated set of terms in a candidate snippet S in such way that maximizes QueryTerm-Keyword Similarity.

The average similarity between the query and the passage items indicates the ***semantic correlation*** between the two. The query passage semantic correlation values, weighted by the score of their relation type (r) that connects them in WordNet, quantifies the quality of the selected passage. Formally, the ***quality*** of a passage *S* containing *n* terms to some query *q* containing *m* terms is given by:

$$\text{Quality}(S, q) = \frac{1}{n \times m} \sum_{j=1}^{m} \{ \sum_{k=1}^{n} [\text{Similarity}(q_j, S_k) \bullet \text{RelationWeight}(r)] \} \quad (7)$$

where, RelationWeights(r) have been experimentally fixed to 1 for *synonymy*, 0.5 for *hypernymy*, *hyponymy* and *has-instance* and 0.4 for *meronymy* and *holonymy*, based on the relation weight values introduced in [32].

The final step towards the identification of the best text fragment within a query matching page is to compute the degree to which a candidate passage makes a useful snippet to the user issuing a query. In measuring the usefulness of a candidate snippet for some query, we rely on the combination of the snippet's relevance and quality to the query intention. Formally, the usefulness of a snippet *S* to a query *q* is:

$$\text{Usefulness}(S, q) = \text{Relevance}(q, S) \bullet \text{Quality}(S, q) \qquad (8)$$

Following the steps described above, in the simplest case, we select from a query matching page the text passage that exhibits the greatest usefulness value to the query intention, as the best snippet to accompany the page retrieved for that query. In a more sophisticated approach, we could select more than one useful passages and merge them in a coherent and expressive snippet.

2.3 Towards Coherent and Expressive Snippets

Having presented our approach towards selecting query–relevant text snippets, we now proceed with the qualitative evaluation of our selection. The aim of our evaluation is to ensure that the snippets presented to the user are both coherent and text-expressive. By coherent, we mean that the selected snippet is well-written and meaningful to the human reader, whereas by text-expressive we mean that the selected snippet represents the semantics of the entire document in which it appears.

Snippet coherence is important in helping the users infer the potential usefulness of a search result before they actually clicks on that. Snippet expressiveness is important after the user clicks on a snippet, since it guarantees that the snippet is representative of the target page content. Given that our passage selection method operates upon the semantic matching between the query intention and the snippet terms, the evaluation of a snippet's coherence focuses on semantic rather than syntactic aspects. That is, in our evaluation we measure the degree to which terms within the snippet semantically relate to each other. To evaluate semantic coherence of a selected snippet, we map all its content terms to their corresponding WordNet nodes. Thereafter, we apply the Wu and Palmer similarity metric (cf. Section 2.1) in order to compute the degree to which snippet terms correlate to

each other. Based on the average paired similarity values between snippet terms, we derive the degree of the in-snippet semantic coherence as:

$$\text{Coherence}(S_1) = \frac{1}{n} \sum_{i,j=1}^{n} \arg\max_{w_j} \text{similarity}(w_i, w_j) \qquad (9)$$

where Coherence denotes the ***in-snippet semantic correlation*** of the *n* terms of snippet S_1. Since the appropriate senses for words w_i and w_j are not known, our measure selects the senses which maximize Similarity (w_i, w_j).

Measuring semantic coherence translates into quantifying the degree of semantic relatedness between terms within a passage. This way, high in-snippet average similarity values yield semantically coherent passages. Semantic coherence is a valuable indicator towards evaluating the degree to which a selected passage is understandable by the human reader. However, even if a passage is semantically coherent, there is no guarantee that the information it brings is expressive of the document's entire content.

Snippet ***expressiveness is*** the degree in which a selected passage is expressive of the entire document's semantics. To quantify the text-expressiveness of a selected passage we want to compute the terminological overlap and the semantic correlation between the selected passage and the rest of its source text. Our computational model is analogous to the query-snippet usefulness metric with the only difference that in this case we compare sets of passages rather than sets of keywords.

More specifically, we take all the content terms inside a document (selected snippet's content terms included), we map them to their corresponding WordNet nodes and we define the *Expressiveness* of a snippet (S_1) in the context of document *D* as follows:

$$\text{Expressiveness}\left(S_1, \left(D - S_1\right)\right) = \text{Usefulness}\left(S_1, \left(D - S_1\right)\right) \qquad (10)$$

where Usefulness (S_1,(D-S_1)) denotes the product of (i) the terminological overlap (i.e. Relevance) between the terms in the selected snippet and the terms in the remaining source document (i.e. D-S1) and (ii) the average semantic correlation between the passage and the remaining text items, weighted by their Relation (r) type.

Based on the above formula, we evaluate the level of expressiveness that a selected passage provides to the semantics of the entire text in which it appears. The expressiveness of a snippet increases proportionally to the amount of the semantically

correlated terms between the snippet and the rest of the text in its source document. The combined application of the snippet coherence and expressiveness metrics gives an indication on the contribution of a snippet in conveying the message of a document retrieved in response to a user query.

2.4 Combination of Metrics for Snippet Selection

So far, we have described our technique for selecting, from a query matching document, the fragments that are semantically close to the query intention. Moreover, we have introduced qualitative measures for assessing how comprehensive is the selected snippet to the human reader and how expressive it is of the entire document semantics.

We now turn our attention on how we can put together the criteria of usefulness; semantic coherence and text expressiveness, in order to assist users perform focused web searches. The foremost decision is to ***balance the influence of each criterion*** in our final selection on the best snippet. In other words, we need to decide whether a query-useful snippet should be valued higher than a semantically coherent or text expressive snippet and vice versa.

Apparently, the weight that could or should be given to each of the individual scores cannot be easily determined and even if this is experimentally fixed to some threshold, it still bears subjectivity as it depends on several factors such as the characteristics of the dataset, the user needs, the nature of the query and many other.

One way to go about determining which of the three metrics (i.e. Usefulness, Coherence and Expressiveness) should be preferred in the snippet selection is to account for the terms' overall importance inside the snippets that each of the above measures delivers. If the three metrics are in agreement (i.e. they deliver the same snippet), the choice is straightforward. However, when they disagree (i.e. select different snippets for the same document and query), we need to identify the winning snippet of the three. In our current study, we rely on the intuition that the most valuable snippet is the one that contains the most important terms. For measuring the importance of terms inside the snippets extracted by the different selection criteria, we rely on the TF-IDF values of the snippet terms, previously computed (cf. Equation 1). We then sum the weights of the terms inside a snippet in order to derive the snippet score. In rough terms, the score of a snippet S containing n terms, each of weight w_i is given by:

$$\text{Score}(S) = \sum_{i=1}^{n} w_i \qquad (11)$$

Depending on the metric that systematically delivers the most highly scored snippets, we can determine the contribution of that metric in the snippet selection process. For instance, assuming that we apply each of the SemSS metrics to a number of documents and queries and assuming also that every metric computes different snippets from the same set of pages; we can rely on the snippets' scores in order to determine the degree to which each of the three criteria contributes in the final snippet selection. That is, in case the Usefulness metric delivers the highest scored snippets from say 70 of 100 documents considered, then the contribution of Usefulness will be fixed to 0.7 in the final snippet selection. Likewise, if Coherence returns the highest scored snippet in 10 of the 100 documents, then its contribution in the final selection will be fixed to 0.1. Analogously, if Expressiveness delivers the highest scored snippets in the remaining 20 documents, its contribution will be 0.2. Based on the above and after several iterations on a large dataset, we can determine the threshold value that should be given to each of the criteria when combining all three of them in the final snippet selection.

Another approach is to present multiple snippets from each document in the query results (i.e. the best snippet when accounting only one criterion each time) and consequently exploit user feedback to conclude on how users perceive the contribution that different values have on snippet-driven retrieval performance. Based on the users' implicit feedback about what makes a good snippet, we could determine the most appropriate weighting scheme for every user [19].

2.4.1 Practical Considerations. A critical issue, concerns the ***visualization of the selected snippets*** to the end user. We claim that it would be useful to highlight the query terms and their synonyms inside the selected snippets, so that the users can readily detect their search targets. Moreover, it would be convenient that text passages are clickable and upon their selection they direct the users to the query relevant snippet rather than the beginning of the document. This way, we can take off the users the burden of reading through the entire document until they detect the information that is most relevant to their query intentions. The snippet selection process can be enriched by merging together snippets from multiple documents and by presenting the merged snippet to the users as an extended answer to their information interests.

In overall, deciding on what makes a good snippet for a particular user information need is a challenge that leaves ample space for discussion and experimentation. Next, we

present an experimental study that we conducted in order to validate the contribution of SemSS method in focused retrieval performance and we discuss obtained results.

3. EXPERIMENTAL EVALUATION

In the previous sections, we have proposed several metrics to select snippets from the contents of the query matching pages. In this section, we experimentally evaluate the effectiveness of these metrics in order to derive perceptible evidence about the impact that the SemSS model has on focused retrieval performance. In Section 3.1, we describe the dataset that we used for our evaluation. In Section 3.2, we discuss how we evaluated the effectiveness of our model. In Section 3.3 we represent obtained results. In Section 3.4 we describe a human study we carried out and we discuss obtained results, which demonstrate the usefulness of the snippets delivered by our model. Finally, in Section 3.5, we compare the performance of the SemSS algorithm to the performance of existing passage selection algorithms, which mainly rely on the statistical distribution of terms within text and we discuss obtained results.

3.1 Experimental Dataset

For our experiments, we downloaded a total set of 590,000 web pages from Google Directory[4] that span 167 distinct topical categories. We firstly processed the downloaded pages in order to remove markup and eliminate pages of non-textual content (i.e. pages containing audiovisual data, links-only, frames, etc.). We then relied on the 538,000 pages of textual content in our dataset, which we tokenized, POS-tagged[5] and lemmatized. Having determined the set of experimental pages against which we would evaluate the efficiency of our SemSS model, the next step was to identify a set of candidate queries that would retrieve relevant documents from our collection. In picking our experimental queries, we decided to rely on the documents' important terms (single words and multiword phrases) based on the intuition that an important document term would be informative of the document's entire content.

For important terms' extraction, we worked as follows: we used the LingPipe[6] Named Entity Recognition software on our experimental documents and extracted a set of named entities from our dataset. We then measured the distribution of the extracted named

[4] http://directory.google.com

[5] Using the Brill POS-tagger

[6] http://www.alias-i.com/lingpipe

entities across the documents in our collection and retained those entities that appear in at least 10 different documents. We also used the Yahoo Term Extraction[7] web service in which we supplied our experimental documents and it returned a list of important words and/or phrases extracted from each of the Google Directory documents in our collection. Having compiled a list of informative terms for each of our experimental pages, we randomly picked a set of 60 important terms as our experimental queries. The only criterion that we applied in selecting our experimental queries was that the selected terms should span the following types: named entities, single words and multiword phrases. The distribution of queries in the above categories is: 10 named entities queries, 35 single term queries and 15 multiword phrase queries. Based on this list of sample queries and the set of the 538,000 processed Google Directory documents, we evaluated the effectiveness of our SemSS model as follows.

We submitted each of the experimental queries to the Google Directory web search service and obtained the pages retrieved for each of the queries. From all the pages that Google Directory retrieved for every query, we retained only those that appear in our experimental dataset (i.e. the 538,000 web pages out of which we have extracted our experimental queries). We then picked from the query relevant pages that Google Directory delivers and which are also present in our experimental dataset only ten pages for each of our queries. This way, we ended up with a total set of 600 web pages (10 pages per query). Since we didn't have pre-annotated queries and results, recall, precision, f-measure and other typical evaluation metrics of information retrieval, could not be applied in our case. Therefore, we adopted a process similar to the one described in [46].

For each of the ten pages retuned for every query, we relied on human judgments about which document fragment is the most helpful to the users for making clicking decisions. To carry out our study, we recruited 10 postgraduate students from our school, who were all experienced web searchers with high levels of computer literacy and who also volunteered to participate in our study. While recruiting our subjects, we told them that their task would be to read a number of pages that are relevant to a set of search queries and indicate in the contents of every page two or three sentences that in their opinion are the most informative about the query-document correlation. In simple words, we presented to our participants the list of our experimental queries and the ten pages retrieved for each of the queries and asked them to identify within the pages' contents two or three sentences that they considered as the most relevant to the query intention.

[7] http://www.programmableweb.com/api/yahoo-term-extraction /

Note that our participants were not aware of our study's objective and they were told nothing about the selection criteria of both the pages and the queries. To assist our participants decipher the intentions of every query, we disambiguated our experimental queries, based on the method described in Section 2.1[8]. Following disambiguation, we presented to our participants the list of experimental queries annotated with their identified senses. That is, every query was tagged with the respective WordNet sense that our disambiguation formula identified as the most indicative of the query's intention. Based on sense annotated queries, we believe that our subjects would be better able to contribute in our study. Recall that since our query disambiguation method operates upon the semantics of the query relevant (i.e. retrieved) documents, the senses appended to each of the experimental queries correspond to the concepts that query terms represent in the respective pages.

Based on the set of disambiguated queries and their retrieved pages, our subjects were instructed to pick from every page contents two or three sentences that were in their opinion the most useful to the query intention (i.e. most relevant to the underlying query sense). Moreover, we advised our participants to freely decide upon the sentence(s) they would pick without restricting their selection to: (i) sentences that contain the query terms (i.e. we instructed our subjects that their selected snippets may or may not contain the query keywords), (ii) consecutive sentences in case of multiple sentence selection, or (iii) sentences of a particular length. As a final note, we should mention that our participants were not allowed to communicate with each other during the experiment.

To minimize subjectivity in the human judgments, we distributed the pages retrieved for each of the queries to five different participants. Therefore, every page was examined by five subjects, who had to read its contents and identify the text fragments (i.e. passages) that could be used for focusing retrieval to the intention of the respective query. Note that every page examined by our participants contained on average 12 sentences excluding title, (sub)-section sentences, references and captions. From all the sentences in a page, our subjects had to identify two or three sentences that were in their opinion the most informative of the query intention. However, to eliminate idiosyncratic snippet selections, we considered a snippet to be informative of the document's correlation to the respective query, if at least three of the participants selected the same snippet (set of sentences) as the most informative of the document.

---

[8] Note that although our query disambiguation method discussed in Section 2.1 operates upon the first 20 (N=20) retrieved documents, in our experiment we set the value of N=10.

The final set of snippets that we explored contained 429 text extracts selected from 429 documents that are retuned for a total number of 54 queries. The distribution of those queries to the categories considered is as follows: 10 named entities queries, 31 single term and 13 multiword queries. These numbers indicate that for 6 out of the 60 experimental queries, our subjects indicated totally different text extracts as informative of the query matching document contents, while for the remaining 54 queries at least three of our participants agreed on a snippet in nearly 8 of the 10 documents that they examined.

The next step is to measure (i) how many of the manually selected snippets were also identified by our techniques, (ii) the accuracy of our method in selecting informative snippets, as well as (iii) the efficiency of our method in delivering informative snippets.

3.2 Evaluation Measures

For our evaluation, we supplied to our model the 54 disambiguated experimental queries and their respective 429 relevant documents. SemSS, following the process described above, extracted a number of snippets from the contents of every query relevant page. The system-selected snippets were extracted in terms of their usefulness, coherence and expressiveness values that SemSS estimated. Then, we evaluated the snippets that our model delivered by comparing them to the snippets that our participants manually determined for the same set of pages and queries. Note that the manually selected snippets considered in our evaluation are those selected by the majority (i.e. 3 out of 5) of the participants who examined them. We started our evaluation by estimating the recall and the precision of the system selected snippets.

To evaluate recall, we rely on the overlapping elements between the set of manually selected snippets for each of the query matching pages examined and the set of system selected snippets for the same set of pages and queries. Note that both sets are of size k, where k is the total number of snippets examined (k = 429). We define recall as the fraction of the manually selected snippets that are also extracted by our model (i.e. automatically selected snippets), formally given by:

$$\text{recall} = \frac{\left|\{\text{manually\_selected\_snippets}\} \cap \{\text{automatically\_selected\_snippets}\}\right|}{k} \quad (12)$$

where the nominator is the size of the intersection of the two sets of snippets and k is the total number of snippets considered.

To estimate recall for the different snippet selection metrics, we measured how many of the manually selected snippets are also identified by: (i) each of our model's metrics (i.e. usefulness, coherence and expressiveness), and (ii) the combination of all snippet selection metrics.

We also evaluated the precision of the snippets selected by our model for each of the examined pages and queries, using the same methodology that we used for estimating recall. In particular, given that our model selected snippets that our subjects did not indicate as informative when marking the informative sentences in the documents, we wanted to examine whether these snippets can help our subjects infer the correlation that their source pages have to the query intention.

More specifically, to evaluate precision, we asked our participants to examine the snippets extracted by each of our selection metrics as well as by their combinations and assess the following: (i) whether the snippet is helpful in deciding whether to click on the respective document, (ii) whether the correlation between the query intention and the retrieved document can be inferred by the displayed snippet. In our evaluation, we consider a snippet to be *precise* if both criteria are met. Therefore, we define precision as the ratio of precise snippets over the total number of automatically selected snippets (i.e. k = 429), given by:

$$\text{precision} = \frac{|\text{precise\_snippets}|}{k} \tag{13}$$

As in the case of recall measurement, each snippet was examined by five different subjects and we considered a snippet to be precise if at least three of the participants marked the snippet as being precise.

As a final evaluation measure, we considered the efficiency of our model in selecting snippets that are valuable to web information seekers. That is, we measured the average response time of our model between issuing a query and selecting the snippets to be displayed together with search results. Average response time includes the time our model requires for processing the query relevant documents in order to disambiguate the query intention ($t_{query\_disambiguation}$) as well as the time our model needs for selecting from the query relevant documents the text fragment ($t_{snippet\_selection}$) that is most likely to help search engine users decide whether to click on a page. Formally, we estimate our model's response time as:

$$\text{Response time} = t_{query\_disambiguation} + t_{snippet\_selection} \quad (14)$$

Based on the above, we evaluate our model's efficiency so that the less time our approach requires for selecting informative snippets, the more efficient it is. Note that the snippet selection time in the previous formula depends not only on the nature of the query and the query relevant pages but also on the different criteria employed for the snippet selection process; in the sense that the combination of all three metrics needs more computations and as such it increases our model's response time and deteriorates efficiency.

In the following section we report obtained results and we discuss our model's performance in delivering informative snippets in terms of precision, recall and computational efficiency.

3.3 Experimental Results

3.3.1 Recall. Table I lists the performance of SemSS model with respect to the measurement of recall. In particular, the table reports the recall values that each of our snippet selection metrics demonstrates as well as the recall of the metrics' combination.

Table I. Recall of the Selected Snippets as Judged by Users

Selection Criteria	Recall for Different Query Types			
	Named Entities	Single Term	Multi Word	All
Usefulness	0.80	0.73	0.64	0.72
Coherence	0.61	0.49	0.31	0.47
Expressiveness	0.72	0.50	0.52	0.58
Usefulness & Coherence	0.83	0.74	0.69	0.75
Usefulness & Expressiveness	0.85	0.78	0.70	0.78
Coherence & Expressiveness	0.79	0.79	0.69	0.76
All	**0.91**	0.84	0.78	0.84

In general, recall improves as we increase the number of features considered for snippets' selection. The criteria of Coherence and Text Expressiveness, when considered individually, tend to perform relatively poor compared to the performance of the Usefulness criterion. This practically implies that the main expectation of the users from the displayed snippets is that they are descriptive of their correlation to the query intention rather than they are representative of the documents' semantics. Another notable observation is that for named entity queries, the criterion of Usefulness in the

snippet selection process is largely sufficient for selecting an informative snippet from a query relevant document.

Therefore, we may assume that for named entity queries we can solely rely on the terminological overlap between the query terms and the page's contents as well as on the semantic correlation between the two. On the other hand, for multi-word queries recall is increased when multiple factors are considered in the snippet selection process (i.e. all explored metrics). In overall, results indicate that the SemSS model has a significant potential in selecting informative snippets from the query relevant documents.

3.3.2 Precision. However, recall is not a sufficient indicator of our model's accuracy in selecting precise snippets, i.e. snippets that are informative of their correlation to the query and which assist users make informed clicking decisions. To capture our model's accuracy in extracting precise snippets we asked our subjects to assess the snippets selected by our model's metric from the pages retrieved for our experimental queries (cf. Section 3.2). We list precision results in Table II.

Table II. Precision of the Selected Snippets as Judged by Users

Snippet Selection Criteria	Precision for Different Query Types			
	Named Entities	Single Term	Multi Word	All
Usefulness	0.80	0.83	0.78	0.80
Coherence	0.69	0.73	0.71	0.71
Expressiveness	0.43	0.64	0.54	0.54
Usefulness & Coherence	0.84	**0.95**	0.89	0.89
Usefulness & Expressiveness	0.89	0.85	0.80	0.85
Coherence & Expressiveness	0.80	0.84	0.77	0.80
All	0.89	0.81	0.84	0.85

As we can see from the table, precision generally increases as more criteria are considered in the snippet selection process. In particular, the combination of multiple selection metrics yields improved precision in almost all of the snippets extracted from the pages retrieved for different query types. The highest precision snippets are those selected by the combination of Usefulness and Coherence metrics and this is mainly pronounced for the documents retrieved in response to single term queries. Obtained results, essentially augment our finding that users prefer snippets that are informative of the correlation between the retrieved pages and their query intention and also imply that users value the well-written, well-organized and coherent snippets as more informative

than others. Moreover, results validate our intuition that well-written and query useful snippets can help web users make informed clicking decisions in focused web searches.

On the other hand, snippet selection based on the criterion of text expressiveness alone results into decreased precision compared to the performance of the other criteria. This intuitively indicates that users value higher the contribution of snippets before rather that after clicking on a retrieved snippet. However, this last observation merits further investigation before can safely rely on it. In overall, results demonstrate that our semantics-driven snippet selection model can accurately identify informative text fragments within the contents of search results and as such it can approximate human judgments about what makes a text snippet helpful, informative and descriptive of web search results.

3.3.3 Efficiency. Another factor we examined is our model's efficiency, i.e. the amount of time our technique requires for selecting precise text snippets from the query relevant documents. In our experiments, the query identification process took on average 28 seconds per query and the main bottleneck was the time needed for pages' pre-processing, keywords' extraction and computation of paired semantic similarity values. When we pre-process the document collection, extract and index their keywords, our method needs on average 6 seconds for identifying the intention of a query. In a real deployment though where the same query might be issued more that once and where web pages' pre-processing is performed at the index level and not a query time, we can speed up the query disambiguation process by storing the identified senses for popular queries offline and use pre-computed values for enabling on-the-fly query sense resolution.

Having identified the query intention and relying on the processed query relevant web pages, our model needs approximately 21 seconds for selecting useful, coherent and expressive snippets from the contents of our 429 experimental pages. This practically means that our model needs on average 0.04 seconds for selecting useful, coherent and expressive snippets from the contents of the first 10 pages retrieved for a query. The snippet selection time is slightly reduced to 0.03 seconds on average for the first ten pages retrieved for a query when considering only one selection metric, i.e. usefulness, coherence or expressiveness. As mentioned before, in a real deployment scenario we can pre-compute snippets for particular queries and their returned pages offline and ensure that results are instantly delivered to web information seekers. In overall, our findings demonstrate our model's efficiency in selecting informative snippets from the query

relevant pages and as such show that our approach has a good potential in assisting web users focus their searches on useful information.

3.4 User Study

Having accumulated perceptible evidence on the effectiveness of our semantically-driven snippet selection approach, we evaluated the impact of the snippets delivered by the SemSS model to the users' clicking decisions. For our evaluation we worked as follows. We issued our 54 experimental queries to the search mechanism of Google Directory and we collected the pages retrieved for each query and their respective text snippets that Google computes based on statistical features (i.e. the distribution of query terms within the matching pages' contents). Recall that the same set of pages have been previously processed by our model and employed for assessing the performance of our semantically-driven snippet selection technique.

The statistically generated snippets from Google Directory and the snippets extracted from the same pages by our semantically-based selection model along with the respective disambiguated queries were stored into a local RDBMS. The information that our database maintains is in the form: <*query*, *query-intention*, *statistical-snippets*, *semantic-snippets*>. Note that for every query, we maintain in our repository 10 statistical and 10 semantic snippets, one for each of the top ten retrieved pages. The best snippet is extracted from each page using both techniques (Google's and ours) so that snippets do not necessarily match. Note also that semantically-generated snippets, delivered by our model, are determined from the combination of all three selection criteria suggested, i.e. usefulness, coherence and expressiveness.

Based on the above dataset, we implemented a simple interface so as to display our experimental queries, their intentions and the set of snippets (statistical and semantic) that are extracted from their matching pages. We then relied on our 10 study participants, asked them to examine the displayed information and make clicking decisions as follows. Each of our participants was shown the full list of queries and, after clicking on a query, the query intention (i.e. WordNet sense) was displayed in a separate field on the interface. Moreover, together with the query intention there were also displayed two lists of text passages; the first list containing the statistical snippets and the second list containing the semantic snippets that have been extracted from the contents of the ten query matching pages. Note that the snippet's ordering adheres to the relative ordering that their corresponding pages have in Google Directory's search results for the given query. That is, the snippets selected from the page that shows up higher in the results' list

from all the ten pages considered will be ranked first among the snippet pairs displayed for the respective query. In case the two snippet extraction techniques (i.e. Google and our model) delivered the same snippet for a query relevant page, the snippet was displayed in both lists. As a result, the snippets of the two lists were presented side by side and were directly comparable.

We then asked our participants to examine the displayed snippet pairs, for as many queries as possible, and indicate for each pair the snippet that would most likely direct them to a document that can successfully answer the query intention.

Our subjects were instructed to indicate their selections by clicking on the respective snippet (only one of the two snippets should be clicked) and make a selection only if they felt confident about that. In other words, we advised our participants that they were not obliged to select one of the two passages in case none of them seemed suitable to the query intention or in case they were unfamiliar with the query intention (i.e. they had difficulty in interpreting the query semantics). A user's click on a snippet translates to a vote given by the user for that snippet's success in focusing retrieval results to the query intention. In case the user clicked on a snippet that was selected by both Google and our model's algorithms, the user's vote was equally attributed to both techniques that delivered the particular snippet. Finally, to ensure that user clicking decisions would be based entirely on the displayed snippets we did not provide any information about the title or the URLs of the pages from which snippets were extracted; information that could influence the users' clicking decisions. For the above reasons, clicking on a snippet would not direct users to the respective pages' contents.

Before we present obtained results, let's first outline some practical issues pertaining to our survey. Out of the 540 pairs of examined snippets (10 snippet pairs per query), 348 were evaluated by at least one participant and 128 of these have been assessed by all our 10 users. In the current work we restrict our evaluation to the snippets with the highest number of annotations; that is we report results for the 128 pairs of snippets that have been assessed by all our study participants. The duration of our experiment was nearly 4 hours and our subjects needed on average 2 minutes for comprehending the query intention and selecting the most suitable snippet from a given pair.

During the experiment's run-time, participants were not allowed to communicate with each other and they were encouraged to take as much time as they needed for making their selections. Note that participants were not told anything about the methods we utilized for selecting the snippets to be displayed and they did not have any internet access during their participation in the experiment. Moreover, to avoid any user bias

towards any of the two snippet selection methods, we displayed the two snippets of a query matching page in either sides of the interface in a random order. Therefore, it is possible that the snippet selected by our model from the first query matching page appears on the left list and the respective snippet for the second query matching page appears on the right etc., thus hiding evidence of the engine that generated each snippet in the pair. Finally, the users were allowed to select the queries and evaluate snippet selection in any order they liked and they were allowed to go back and change their selection at any point during the study.

Table III, shows the results obtained from our human survey for the 128 pairs of snippets that have been assessed by all our study participants. More specifically, the table reports the number of times our subjects clicked on snippets delivered by Google's statistical extraction technique and by our model's semantics-driven method. The table also reports the quantity of overlapping snippets in our dataset, i.e. pairs of identical snippets delivered by both methods.

Table III. Distribution of User Preferences between the Snippets Extracted from the Same Set of Pages and Queries by Google's Statistical Algorithm and our Model's Semantically-Driven Approach respectively

Distribution of user clicks	
Number of clicks	128
Clicks attributed to statistical snippets	37
Clicks attributed to semantic snippets	81
Clicks attributed to overlapping snippets	10

As we can see from the table, the majority of the user clicks are directed to snippets delivered by our semantically-driven selection technique. Obtained results suggest that the snippets delivered by our model are valued higher for focusing web searches, compared to the snippets delivered by statistical methods that most search engines currently employ. This observation is in line with our intuition that snippet selection based on the semantic, rather than the statistical properties of documents, yields improved web searches in terms of the users' clicking decisions on the lists of search results.

Another observation is that 10 out of the 128 snippet pairs that our subjects evaluated contained identical snippets. This practically means that for 10 pages in our collection, the two algorithms extracted from their contents the same text fragments as the most

relevant to the respective query intention. Based on the above, we can derive the degree of selection agreement between the two algorithms, which in our experiment is roughly 8%. This number indicates that there is a great difference in the way our model selects snippets from the query matching pages compared to the snippet selection technique that Google search engine adopts.

Overall, the results of our user study indicate that users prefer semantically-driven snippets, and rely their clicking decisions upon them, especially when they have crystallized their search focus, i.e. they have a specific information need in mind. Furthermore, we believe that by exploring semantically-generated snippets the users can locate pages of interest faster, without the need to actually visit search results in order to conclude whether a page suits their search intention or not.

Based on our findings so far, we claim that the search engine community can benefit from more sophisticated approaches towards snippets' selection and we believe that the exploitation of semantically-driven passage selection algorithms can be fruitfully employed in this respect. Our suggested snippet selection model can extend existing algorithms to work over the integration of statistical and semantic features, allowing web information seekers to quickly discover interesting pages within search results.

3.5 Comparison with Baseline Algorithms

In this section we quantitatively compare the performance of the SemSS model to the performance of existing baseline passage retrieval algorithms. Passage retrieval has been insofar addressed in the context of question-answering applications and aims at targeting information retrieval. That is, instead of retrieving a list of potentially relevant documents, it extracts from the relevant documents' contents the information requested by the user. Given that the focus of SemSS model is the extraction of query-useful text fragments from the query-relevant pages, we believe that our approach has a common objective with passage retrieval methods: to identify within a query-relevant document the text nugget (typically paragraph-sized) that better matches the query intention.

For our study, we implemented two different passage retrieval algorithms that have a high performance in TREC-10 systems and which are well-described in the literature. The algorithms that we employed as the baseline passage retrieval methods for our study are:

- **MultiText Algorithm**. The MultiText algorithm [7][6] is a density-based passage retrieval algorithm that favors short passages containing many terms with high IDF values. Each passage window in the algorithm starts and ends

with a query term, and its score is based on the number of query terms in the passage as well as in the window size. The highest scoring passage is the one deemed as most probable for satisfying the query intention.

- **IBM Algorithm**. The IBM's passage retrieval algorithm [16] relies on the linear combination of six different features encountered in the candidate query-useful passages. The features considered are: the *matching words measure*, which sums the IDF values of words that appear in both the query and the passage. The *thesaurus match measure* that sums the IDF values of words in the query whose WordNet synonyms appear in the passage. The *mismatch words measure*, which sums the IDF values of words that appear in the query and not in the passage. The *dispersion measure* that counts the number of words in the passage between matching query terms. The *cluster words measure*, which counts the number of words that occur adjacently in both the query and the passage. Based on the linear combination of the above scores, the algorithm picks the passage that exhibits the highest sum score as the most probable for satisfying the query intention. In our implementation of the algorithm, we determine the candidate query passages using a sliding window of 35 consecutive terms around the query-matching keywords.

Having presented the baseline passage retrieval algorithms against which we evaluate the performance of SemSS model, we proceed with the description of our experimental setup. For our evaluation, we relied on our experimental data (cf. Section 3.1) from which we picked a set of 12 queries and their corresponding 10 pages retrieved from Google Directory. In selecting the queries, we explored the list of the 54 queries for which at least three of our subjects indicated the same snippet in the contents of the query matching pages as being informative of the document's correlation to the query intention. From those queries, we picked the ones for which at least five different subjects selected the same text fragments from each of their ten retrieved pages as being informative snippets of the query-page correlation. This way, we ended up with a total set of 12 queries for which five users agreed on the same snippet in each of the ten pages considered for the queries.

Having collected our experimental dataset, we run each of the three algorithms considered (i.e. MultiText, IBM and SemSS) on the first ten pages that Google Directory returns for each of our 12 experimental queries and which are also represented in our experimental collection. Each algorithm extracted the best passage from every document and returned ten passages for each of our experimental queries. Note here that the IDF

values in the matching pages' terms, upon which both baseline algorithms operate, had already been computed while preprocessing our dataset (cf. Section 3.1) for disambiguating the query intention. Therefore, there was no need for any additional documents' pre-processing in order for the baseline algorithms to compute a query-relevant text fragment.

Our implementation ignored the original document ranking and identified the best passage from every document based on the underlying scoring function that each of the algorithms incorporates. Note that the snippet scoring function that our model uses in the reported implementation is the one given by Equation 11 and is parameterized so as to consider only the usefulness metric in the snippet selection process. The reason for examining only the criterion of usefulness is practically in order to ensure that all algorithms considered operate upon comparative measures for snippets' selection, i.e. the query-passage correlation. After running the three algorithms for the same set of documents and queries, we retained from each of the pages examined the passage with the highest score for each of the algorithms.

Given that the algorithms we explored differ in functionality, the passages that they select varied in length. Therefore, we normalized the extracted snippets, by either expanding or reducing their size so as to fit within a window of 35[9] word tokens. In other words, in case a passage extracted by an algorithm contained fewer that 35 words, we expanded it on both ends by adding words that surround the passages in its source document. Conversely, if a passage returned by an algorithm contained more than 35 words, we trimmed redundant words from both ends.

Having collected and normalized the highest scored snippets that each of the algorithms computed from each of the 120 pages (10 pages per query), we evaluated the impact of the different passage retrieval methods in focused web information retrieval. For our evaluation, we worked as follow: we considered the snippets that our subjects have manually selected from the respective pages' contents as the gold standard query useful snippets. We then compared the coverage ratio of each algorithm in delivering query-useful snippets. For our comparisons, we define coverage ratio as the fraction of the gold standard snippets that are extracted by each of the algorithms. That is, we examined how many of the 120 gold standard snippets are extracted from each of the algorithms for the respective pages and queries.

---

[9] The text window size is fixed to 35 words because this approximates the typical size of text snippets that search engines return together with retrieved results.

Table IV shows the overall retrieval performance of the different algorithms under strict and lenient conditions. A strict condition requires that all the terms in a snippet extracted from a document by a given algorithm are identical to the terms in the gold standard snippet of the document, whereas a lenient condition requires that at least 50% of the terms in a snippet (i.e. 16 of the 35 words) are identical to the terms in the gold standard snippet of the respective document.

Table IV. Retrieval Performance of the three algorithms considered in Extracting Query-Relevant Snippets under Strict and Lenient Evaluation Conditions.

Algorithms	Coverage Ratio	
	Strict	Lenient
MultiText	0.225	0.341
IBM	0.408	0.566
SemSS	**0.608**	**0.733**

Obtained results demonstrate that in overall SemSS yields increased retrieval performance compared to the performance of existing passage retrieval techniques. In particular, we observe that our algorithm managed to select the most informative (i.e. gold standard) snippet under strict conditions in 73 out of the 120 documents examined, which practically translates into an overall effectiveness of 60.8%. Conversely, the performance of the two baseline algorithms was significantly lower, which indicates that statistically based retrieval has a limited ability in satisfying the query intention. This is especially true for the MultiText retrieval algorithm that relies mainly on the IDF values of query terms within the documents' contents and considers nothing about the documents' or the query semantics. Moreover, results demonstrate that when lenient evaluation conditions apply (i.e. a partial terminological match between the delivered and the gold-standard snippets is sufficient) the retrieval effectiveness of all the algorithms considered is significantly increased. In particular, retrieval improvement is nearly 11.66% for the MultiText algorithm, 15.83% for the IBM and 12.5% for our model's algorithm. The above figures indicate that performance improvements are analogously pronounced under relaxed evaluation criteria, which essentially demonstrates that obtained results are consistent across different experimental settings.

Overall, results demonstrate that our snippet selection approach that primarily relies on the semantic rather than the statistical properties of terms within the query-relevant documents has a significant potential in delivering useful and informative text fragments

together with search results. As such we believe that our model can assist web information seekers focus their web searches on results that are highly probable of satisfying their search pursuits.

4. RELATED WORK

The role of text snippets or passages in the context of web information retrieval has been studied before. A number of researchers have proposed the exploitation of passages to answer natural language queries [1], [27], [28] and generic queries [9]. Authors in [1] search for single snippet answers to definition questions through the exploitation of lexical rather than semantic patterns. In [27] and [28] the authors exploit WordNet to annotate and consequently answer definition questions. Most of the reported approaches on snippets' exploitation for question-answering rely on some similarity measure in order to derive from a query relevant document, the text fragment that is closest to the query. The relevance/ similarity between the query and the snippet is measured using linguistic [8] (distance in an ontological thesaurus) or statistic [24] (word frequency, proximity or co-occurrence) techniques or a combination of them.

Passage retrieval is a common component to many question answering systems. Currently, there exist several passage retrieval algorithms, such as MITRE [23], bm25 [29], MultiText [7], IBM [16], SiteQ [22], ISI [15]. Recently, [33] quantitatively evaluated the performance that the above passage retrieval algorithms have on question answering. Obtained results showed that passage retrieval performance is greatly dependent upon the document retrieval model employed and that density-based measures for scoring the query terms yield improved rankings to the query relevant passages.

Moreover, passage retrieval approaches have been proposed in the context of web-based question answering [20], [3]. Most of the systems explored in web-based passage retrieval typically perform complex parsing and entity extraction for documents that best match the given queries, which limits the number of web pages that can analyze in detail. Other systems require term weighting for selecting or marking the best-matching passages [6] and this requires auxiliary data structures.

Many research works perform post processing to the snippets extracted from query results. They either cluster snippets into hierarchies [9], use them to construct ontologies [34], or further expand the snippet collection with relevant information nuggets from a reference corpus [25]. Evaluation of retrieved snippets is performed once again using statistic [4]or linguistic methods [35] and long QA series [36]. Text coherence is a topic that has received much attention in the linguistic literature and a variety of both

qualitative and quantitative models have been proposed [12] [14] [10] [21]. Most of existing models incorporate either syntactic or semantics aspects of text coherence.

In our work on passage retrieval, we rely purely on semantic rather than syntactic aspects of both the queries and the documents and propose a novel evaluation framework which ensures that the passage delivered in response to some query is not merely query relevant but also semantically coherent and expressive of the entire document's contents.

Recently, a number of researchers have proposed the exploitation of web snippets for measuring semantic similarity between search keywords. In [30] the authors measured semantic similarity between two queries using the snippets returned for those queries by a search engine. For each query, they collect snippets from a search engine and represent each snippet as a TF-IDF weighted term vector. Each vector is L_2 normalized and the centroid of the set of vectors is computed. Semantic similarity between the two queries is defined as the inner product between the corresponding centroid vectors.

In a similar work, [5] proposed a double-checking model using text snippets returned by a web search engine to compute semantic similarity between words. Experimental results indicate that although two query keywords might be semantically similar to each other, the terms appearing in their retrieved snippets might have low or even zero similarity values. Likewise, [2] recently proposed a robust semantic similarity measure that exploits page counts and text snippets returned by search engines in order to measure the similarity between words or entities. To evaluate their method they ran a community mining and an entity disambiguation experiment. Obtained results demonstrated that their method, relying on web snippets, can accurately capture semantic similarity between named entities.

Although there exist several works that rely on web text snippets in order to improve retrieval performance or to enhance the accuracy of question answering techniques, nevertheless none of the reported attempts tries to improve the snippet selection process in order to assist web users make informed clicking decisions while interacting with a search engine. Our work on the semantically-driven snippet selection contributes to this area and facilitates the deployment of small text fragments extracted from the query retrieved documents in order to focus web searches on the query intentions rather than the query keywords.

This paper substantially expands our previous work in [40]. Our earlier paper presented preliminary results on extracting query useful snippets from search results by relying on manually disambiguated queries. In the current paper, we show how to use the retrieved documents' semantics in order to disambiguate the query intention and we

introduce a novel snippet scoring function that combines the different snippet selection metrics, previously proposed, in order to determine a single text fragment for accompanying retrieval results. Furthermore, we performed extensive user studies in order to evaluate the effectiveness of our snippet selection model and we assessed its performance against existing passage retrieval algorithms.

5. CONCLUDING REMARKS

In this article, we presented a set of semantically-driven techniques for automatically extracting useful text fragments from the contents of the query relevant documents. Our techniques build on the idea that the query intention, when identified with an appropriate disambiguation method, provides useful information for locating the most useful text snippet in a query relevant document and thus it focuses web searches on the user needs. Our approach capitalizes on the notion of semantic correlation between the query keywords and the selected snippet's content as well as on the semantic correlation between the in-snippet terms. We argue that our approach is particularly suited for identifying within the contents of a possibly long document the focus of the query and we introduce a qualitative evaluation scheme for capturing the accuracy in which the selected passage participates in focused web searches.

We applied our snippet selection technique (SemSS) to a number of searches that we have performed using real web data and we compared its performance to the performance of existing passage retrieval algorithms. Experimental results, validated by an extensive study using human subjects, indicate that SemSS approach, which relies on the semantic correlation between the query intention and the retrieved documents' content, identifies text fragments of high quality that as such it can improve the users' search experience.

The snippet selection approach introduced in this article relies on semantic rather than statistical properties of web documents and it is relatively inexpensive assuming access to a rich semantic resource (such as WordNet). This makes the proposed approach particularly attractive and innovative for the automatic selection and evaluation of focused web snippets. We believe that it is relatively straightforward to integrate in our model other semantic resources that are useful in deciphering the intentions of specific queries. For instance, we could employ the Taxonomy Warehouse[10] recourse, which offers a number of domain-specific taxonomies that can be used in identifying specialized terms within the query relevant documents. Thereafter, based on these specialized terms

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

one could judge whether the query intention is focused on a particular subject of interest and based on this knowledge to select the text fragment that best matches the identified intention. For example, when querying the web for information about Java programming language, we can use one of the available glossaries to identify programming-related terms in the contents of search results; thus we can select snippets that are focused to the query intention.

In future research, we plan to enrich the SemSS model with advanced linguistic knowledge such as co-reference resolution, genre detection or topic distillation. Moreover, it would be interested to experiment with alternative formulas for measuring the correlation between the query keywords and the passage terms, such as the one proposed in [13]. Another possible direction would be to employ a query relevant snippet as a backbone resource for a query refinement technique. Yet a more stimulating challenge concerns the incorporation of user profiles in the snippet selection process in an attempt to deliver personalized text passages. Last but not least, our snippet selection approach could be fruitfully explored in the context of web question-answering and element retrieval systems in the hope of helping the user find the exact information sought in an instant yet effective manner.

REFERENCES

[1] I. Androutsopoulos , D. Galanis, A practically unsupervised learning method to identify single-snippet answers to definition questions on the web, in: Proc. HLT/EMNLP Conference, (2005) 323-330.

[2] D. Bollegala, Y. Matsuo, M. Ishizuka, Measuring Semantic Similarity between Words using Web Search Engines, in: Proc. World Wide Web Conference, (2007) 757-766.

[3] S. Buchholz, Using Grammatical Relations, Answer Frequencies and the World Wide Web for TREC Question Answering, in: Proc. 10th TREC Conference (2002)

[4] L.A. Charles, E.L. Clarke, Terra: Passage Retrieval vs. Document Retrieval for Factoid Question Answering, in: Proc. SIGIR Conference, (2003).427-428.

[5] H.Chen, M. Lin, Y. Wei, Novel Association Measures Using Web Search with Double Checking, in: Proc. COLING Conference, (2006) 1009-1016.

[6] C. Clarke, G. Cormack, T. Lynam, Web Reinforced Question Answering, in: Proc. 10th TREC Conference, (2001).

[7] C. Clarke, G. Cormack, D. Kisman, T. Lynam, Question Answering by Passage Selection (Multitext experiments for TREC-9), in: Proc. 9th TREC Conference, (2000).

[8] M. De Boni, S. Manandhar, The Use of Sentence Similarity as a Semantic Relevance Metric for Question Answering, in: New Directions in Question Answering, (2003) 138-144.

[9] P. Ferragina, A. Gulli, A Personalized Search Engine Based on Web-Snippet Hierarchical Clustering, in: Special Interest Tracks & Posters of the 14th International World Wide Web Conference (2005).

[10] P. Foltz, W. Kintsch, K. Landauer, Textual Coherence Using Latent Semantic Analysis, in: Discourse Processes, 25 (2&3): (1998) 285-307.

[11] Google patent 2003. Detecting Query-Specific Duplicate Documents. US patent No. 6615209.

[12] B. Grosz, A. Joshi, S. Weinstein, Centering: A Framework for Modeling the Local Coherence of Discourse, in: Computational Linguistics, 21 (2) (1995) 203-225.

[13] M. Halkidi, B. Nguyen, I. Varlamis, M. Vazirgiannis, THESUS: Organizing Web Document Collections Based on Link Semantics, in VLDB Journal 12(4) (2003) 320-332.

[14] D. Higgins, J. Burstein, D. Marcu, C. Gentile, Evaluating Multiple Aspects of Coherence in Student Essays, in: Proc. NAACL Conference (2004) 185-192.

[15] E. Hovy, U. Hermjakob, C.Y. Lin, The User of External Knowledge on Factoid QA, in: Proc. 10th TREC Conference (2001).

[16] A. Ittycheriah, M. Franz, S. Roukos, IBM's Statistical Question Answering System-TREC10, in: Proc. 10th TREC Conference (2001).

[17] B.J. Jansen, A. Spink, T. Saracevic, Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web, in: Information Processing & Management, 36 (2) (2000) 207-227.

[18] E. Aggire, Ph. Edmonds, Word Sense Disambiguation: Algorithms and Applications, Springer (Dordrecht, 2006).

[19] A. Kritikopoulos, M. Sideri, I. Varlamis, Success Index: Measuring the Efficiency of Search Engines Using Implicit User Feedback, in: Proc. 11th Pan-Hellenic Conference on Informatics, Special Session on Web Search and Mining (2007).

[20] C. Kwok, O. Etzioni, D. Weld, Scaling Question Answering to the Web, in: Proc. 10th World Wide Web Conference (2001).

[21] M. Lapata, R. Barzilay, Automatic Evaluation of Text Coherence: Models and Representations, in: Proc. International Joint Conferences on Artificial Intelligence (2005).

[22] G.G. Lee, J. Seo, S. Lee, H. Jung, B.H. Cho, C. Lee, B.K. Kwak, J. Cha, D. Kim, J. An, H. Kim, K. Kim, SiteQ: Engineering High Performance QA System Using Lexico-semantic Pattern Matching and Shallow NLP, in: Proc. 10th TREC Conference (2001).

[23] M. Light, G.S. Mann, E. Riloff, E. Breck, Analyses for Elucidating Current Question Answering Technology, in: Journal of Natural Language Engineering, Special Issue in Question Answering (2001).

[24] D. Lin, P. Pantel, Discovery of Inference Rules for Question Answering, in: Natural Language Engineering 7 (4) (2001) 343-360.

[25] G. Mishne, M. de Rijke, V. Jijkoun, Using a Reference Corpus as a User Model for Focused Information Retrieval, in: Journal of Digital Information Management, 3 (1) (2005) 47-52.

[26] T. Pedersen, S. Banerjee, S. Patwardhan, Maximizing semantic relatedness to perform word sense disambiguation, University of Minnesota Supercomputing Institute, Research Report UMSI 2005/25 (2005).

[27] J. Prager, J. Chu-Carroll, K. Czuba, Use of WordNet Hypernyms for Answering What-Is Questions, in Proc. TREC-2002 Conference (2001).

[28] J.M. Prager, D.R. Radev, K. Czuba, Answering What-Is Questions by Virtual Annotation, in: Proc. of Human Language Technologies Conference (2001) 26-30.

[29] S.E. Robertson, S. Walker, M. Hancock-Beaulieu, M. Gatford, A. Payne, Okapi at TREC-4, in Proc. 4th TREC Conference (1995).

[30] M. Sahami, T. Heilman, A Web-Based Kernel Function for Measuring the Similarity of Short Text Snippets, in Proc. 15th International World Wide Web Conference (2006).

[31] G. Salton, C. Buckley, Term Weighting Approaches in Automatic Text Retrieval, in: Information Processing and Management, 24 (5) (1988) 513-523.

[32] Y. I. Song, K.S. Han, H.C. Rim, A Term Weighting Method Based on Lexical Chain for Automatic Summarization, in: Proc. 5th CICLing Conference (2004) 636-639.

[33] S. Tellex, B. Katz, J. Lin, A. Fernandes, G. Marton, Quantitative Evaluation of Passage Retrieval Algorithms for Question Answering, in Proc. 26th SIGIR Conference (2003).

[34] W.R. Van Hage, M. de Rijke, M. Marx, Information Retrieval Support for Ontology Construction and Use, in Proc. International Semantic Web Conference (2004) 518-533.

[35] E. Voorhees, Evaluating Answers to Definition Questions, in: Proc. HLT-NAACL Conference (2003).

[36] E. Voorhees, Using Question Series to Evaluate Question Answering System Effectiveness, in Proc. HLT/EMNLP Conference (2005).

[37] WordNet. available at: http://www.cogsci.princeton.edu/~wn

[38] X. Wu, M. Palmer, Web Semantics and Lexical Selection, in: Proc. 32nd ACL Meeting (1994)

[39] O. Zamir, O. Etzioni, Web Document Clustering: A Feasibility Demonstration, in Proc. SIGIR Conference (1998).

[40] N. Zotos , P. Tzekou, G. Tsatsaronis, L. Kozanidis, S. Stamou, I. Varlamis, To Click or not to Click? The Role of Contextualized and User-Centric Web Snippets, in: Proc. SIGIR Workshop on Focused Retrieval (2007)

[41] E. Agirre, P. Edmonds, Word Sense Disambiguation: Algorithms and Applications, in Text, Speech and Language Technology, 33 (Springer, 2007)

[42] G. Tsatsaronis, M. Vazirgiannis, I. Androutsopoulos, Word Sense Disambiguation with Spreading Activation Networks Generated from Thesauri, in: Proc. 20th International Joint Conference on Artificial Intelligence, IJCAI (2007) 1725-1730.

[43] R. Krovetz, W. Croft, Lexical ambiguity and information retrieval, in: ACM Transactions on Information Systems, 10 (2) (1997) 115-141.

[44] R. Krovetz, Homonomy and polysemy in information retrieval, in: 35th Meeting of the Association for Computational Linguistics and the 8th Meeting of the European Chapter of the Association for Computational Linguistics (ACL/EACL-97), (1997) 72-78.

[45] A. Budanitsky, G. Hirst, Evaluating wordnet-based measures of lexical semantic relatedness, in: Computational Linguistics, 32 (1) (2006) 13–47.

[46] W. Dakka P. Ipeirotos, Automatic extraction of Useful facet Hierarchies from Text Databases, in: Proc. 24th IEEE International Conference on Data Engineering (2008).

[47] T. Pedersen, S.V. Pakhomov, S. Patwardhan, C.G. Chute, Measures of semantic similarity and relatedness in the biomedical domain, in: Journal of Biomedical Informatics 40 (3) (2007) 288-299.

[48] G. Tsatsaronis, I. Varlamis, M. Vazirgiannis. Word Sense Disambiguation with Semantic Networks, in: Proc. 11th International Conference on Text, Speech and Dialogue, (2008)..