A Study of Ontology-based Query Expansion

Jiewen Wu, Ihab Ilyas, and Grant Weddell

{j55wu|ilyas|gweddell}@uwaterloo.ca
Cheriton School of Computer Science,
University of Waterloo

Technical Report CS-2011-04

Abstract. With enormous data emerging on the Web, traditional keyword searching is challenged by short queries posed by users to vaguely describe their information need. Query expansion has been researched for decades and a variety of expansion strategies have improved retrieval effectiveness. At present, knowledge-based query expansion approaches are popular as the Web becomes more *semantic*. This paper studies state-of-the-art in ontology-based query expansion approaches, and expands on practical strategies to exploit the rich semantics of domain ontologies. This paper, one the one hand, focuses on finding out the success factors for ontology-based query expansion; on the other hand, it emphasizes the tradeoff between the gained retrieval effectiveness and the incurred computation cost.

1 Introduction

The amount of information available on the Web increases with time, which further advances search services for online users. Searchers naturally prefer to post queries in their native languages, while oftentimes queries in these human languages can not be exactly understood by computers. A simplified and straightforward way to formulate user queries is using keywords in place of natural languages to approximate users' information needs. Keyword queries are then processed by search engines in a Boolean way, i.e., every document is treated as a set element and searching a keyword returns a set of *relevant* documents to the original keywords. Boolean operations, including `AND`, `OR`, `NOT` and so on, are subsequently performed on sets of documents to return the final set of (possibly ranked) documents as results. Discouragingly, up to the present date none of the developments in search services, as far as we know, largely match the search skills of an information specialist, e.g., a librarian, who retrieves specific information pieces strategically and effectively among a vast amount of resources. Bates [1979] studied the human search strategy and compiled a set of tactics that librarians used in information retrieval, in hopes that the tactics can be leveraged in automated search services to help improve search results.

In an online setting, web users tend to follow specific search trends, while present various user behavior in searching. First of all, the majority of user queries carries a short form. For instance, the average number of terms per query is 2.4 for web users, as observed in [Spink et al., 2001], and 1.59 for small-scale system users [Dumais et al., 2003]. Another related characteristic of users is their preference for popular or broad terms, e.g., a number of highly frequent query terms represent sexuality and current news in [Spink et al., 2001]. General terms, more ambiguous than specific ones, result in queries harder for search engines to interpret. Moreover, very few web users adopts advanced searching options, e.g., Boolean operators, in query formulation [Spink et al., 2001; Dumais

et al., 2003]. Regardless of the experience, users tend *not* to refine the queries [Hsieh-Yee, 1993; Spink et al., 2001; Markey, 2007]. A high percentage of mistakes were also observed from users who occasionally employed advanced search options [Spink et al., 2001]. This behavior exhibits a lack of subject knowledge of users to formulate more accessible queries or users expect the search engines to completely satisfy their information needs based on minimal query input.

The challenges posed above can be dealt with in at least two ways: one is to refine the user queries, and the other is to optimize the search service in hopes that the search service return reasonable results whatever the user input is. The orientation of this paper is to focus on refining the user queries, instead of the latter way. Naturally, before initiating a search process, users can be guided to formulate more useful queries in terms of understandability. Structured query processing aims this direction by designing query languages, e.g., database querying in SQL, knowledge base querying in SPARQL, and so on. Nevertheless, end users face insurmountable difficulty in manipulating the query language to formulate keyword queries, especially the web users, in general, posing short and unstructured queries. When queries consisting of a few uncontrolled terms surface, the question becomes how to reformulate user queries into ones that are amenable to search services. This paper thus addresses query expansion (QE), an apparatus to include more relevant search terms in the query for improved retrieval results.

1.1 Query Expansion Overview

Query expansion, generally thought of as a recall-based[1] technique, is aimed to automatically formulate a user query into one that is more amenable for information retrieval. Earlier research [Voorhees, 1994] already showed that though query expansion had limited retrieval improvement on detailed or complete queries, it demonstrated great potential for significantly improving results given short queries.

In literature, QE approaches are studied in different ways. For instance, [Manning et al., 2008, Chap. 9] and [Wollersheim and Rahayu, 2005a] categorized QE approaches into global and local methods, where methods in the first category are query-idependent since all documents are examined for all queries. Conversely, methods in the second class modify a query relative to the documents initially returned by the query. Discussions regarding these approaches can be found in [Xu and Croft, 2000]. Alternatively, Grootjen and van der Weide [2006] characterized QE approaches as extensional, intensional, or collaborative ones. The first set of approaches materializes information need in terms of documents, for instance relevance feedback and local analysis methods. Intensional approaches, primarily thesauri/ontology-based, take advantage of the semantics of keywords. Collaborative ones exploit users' behavior, e.g., mining query logs, as a complement to previous approaches.

This paper refines the classification of mainstream QE approaches as follows. QE approaches are semantic [Aronson et al., 1994; Shah et al., 2002], syntactical (mostly through statistical methods) [Salton and Buckley, 1990; Buckley et al., 1994b; Jing and Croft, 1994], or a combination of the two [Croft, 1986; Farfan et al., 2009], as the syntactic and semantic approaches are not orthogonal. Syntactical approaches considers the term dependence statistically, e.g., exploiting term co-occurrence.

[1] To engance recall or precision depends on the users' objectives. In general, recall is improved as an effect of QE, e.g., [Greenberg, 2001a; Sihvonen and Vakkari, 2004; Egozi et al., 2008], while in some cases precision is enhanced as well, e.g., [Lee et al., 2008].

Semantic term dependence is captured in semantic QE approaches[2]. An overview on statistical approaches follows.

We refer statistical methods, which normally entail global analyses, to the techniques that *understand* a corpus in a statistical manner, mostly based on term co-occurrence analyses. Earlier research (e.g., [Hersh and Hickam, 1995]) strongly suggests the use of statistical methods after their empirical analyses on information retrieval using thesauri. The thesaurus look-up query expansion approach is criticized for causing *query drift* mainly because of the polysemy of words [Buckley et al., 1994b]. However, there also are arguments [Yang and Chute, 1993] supporting that various factors influence the thesauri-based retrieval performance. For instance, a rich vocabulary in a thesaurus and reasonable mapping between different vocabulary, e.g., between thesaurus concepts and textual words, indeed improved the performance [Yang and Chute, 1993; Yang, 1994].

Relevance feedback [Salton and Buckley, 1990], a representative of local analysis approaches, offers benefits for searching small databases, but has a minimal effect in general cases [Hersh and Hickam, 1995]. In fact, relevance feedback suffers from a lack of semantics in the corpus, which restrains its applications in several occasions, for example, when the query concept is as general as a disjunction of more specific concepts [Manning et al., 2008, Chap. 9]. Numerous improvements have been made to local analysis methods Xu and Croft [2000]; Bai et al. [2007]. In particular, Xu and Croft [2000] proposed a local context analysis to combine both global and local analyses. That is, the selection of expansion terms is enhanced by considering concepts in the top-ranked documents that frequently co-occur with "many" query terms in the whole collection. Compared to relevance feedback, candidate expansion terms are more relevant to the query, as they have been observed to co-occur frequently in all documents; hence, such expansion terms minimize the chances of query drift seven if the returned top-ranked documents contain many irrelevant results in the feedback phase. In fact, Xu and Croft [2000] relied on the (reasonable) hypothesis that "a common term from the top-ranked relevant documents will tend to co-occur with all query terms within the top-ranked documents." Note that the above method is more effective if all query terms constitute some *concept* so that they tend to co-occur frequently in the whole collection. A different way to handle irrelevant feedback documents was shown in Mitra et al. [1998]. Focusing on initially retrieving an acceptable set of documents to resolve query drift, Mitra et al. [1998] re-ranked a larger set containing the initial documents to result in a higher proportion of relevant documents used for feedback.

Retrieval feedback [Srinivasan, 1996b], a derivation of relevance feedback, adds terms among the top (either user-selected or pseudo) relevant documents to the query. This suite of relevance-feedback-based approaches has shown significant improvement in many information retrieval tasks. Nevertheless, the effectiveness of feedback-style QE approaches requires a careful selection of the seed queries, ranking functions, and some other factors. Srinivasan [1996b] further seeked to combine the statistical methods with relevance retrieval for query expansion, though the results reflect no or only minimal improvements. Srinivasan [1996b] also confirmed that thesaurus-based methods produce significant improvement. Arguably, Aronson and Rindflesch [1997] concluded that ontology-based QE is a more effective and favorable method than relevance feedback. However, an optimal system that benefits from both retrieval feedback and ontology-based QE is feasible [Srinivasan, 1996c; Aronson and Rindflesch, 1997].

[2] Observe that semantic term dependence can be studied on the document dimension in information retrieval as well, e.g., document expansion or language model smoothing.

1.2 Related Work

Query answering with the use of ontologies has been well researched recently. For instance, ontological information aids in structuring the data [Shah et al., 2002] or query [Pound et al., 2010]. It is rational to believe that search engines may evolve into knowledge engines, because users are becoming more interested in specific facts satisfying the queries while facing a vast volume of returned information. A good illustration is Wolfram|Alpha[3], which returns "systematic factual knowledge" in response to users' free form input. Some query answering engines [Bast et al., 2007; Pound et al., 2010] normally process keyword queries in a structured manner internally over a large ontology to return relevant answers, while most systems accept free text as input, e.g., Wolfram|Alpha, Freebase[4], Kngine[5], etc.

Hoang and Tjoa [2006] surveyed several ontology based query systems on various aspects of using ontologies, including faceted search, query reformulation and refinement and so on, while this paper is specifically devoted to ontology-based query expansion. Bhogal et al. [2007] provided a comprehensive review of ontology based query expansion, which presents several query expansion approaches, focusing on examples using corpus dependent or independent ontologies. Bhogal et al. [2007], however, did not detail how to improve query expansion using ontologies. This paper elaborates on the critical phases of ontology-based query expansion and emphasizes the balance between computation cost and retrieval effectiveness in each phase, based on the analyses of existing works.

1.3 Organization

This paper is organized as follows. The next section, Sect. 2, defines the problem of query expansion and introduces notations used in the discussions throughout this paper. Sect. 3 introduces a synthesized view of query expansion strategies and presents the components of ontology-based query expansion as follows. Sect. 4 elaborates on annotating the underlying corpora with the aid of ontologies. Sect. 5 discusses various aspects of domain ontologies. The core component is Sect. 6 that details the ontology-based query expansion algorithm. The results generated by the core component are processed for robustness, as shown in Sect. 7. Finally, we conclude the paper in Sect. 8.

2 Problem Statement

This section defines the problem and introduces necessary notation. In addition, the motivation of exploiting domain knowledge that is represented in the form of ontologies is discussed in this section.

2.1 Definitions

The problem of query expansion, the definitions of an entity and a document are stated as follows.

[3] http://www.wolframalpha.com
[4] http://www.freebase.com
[5] http://www.kngine.com

Definition 1 (Keyword Query and Query Expansion). *Let* $\mathcal{Q}$, $\mathcal{C}$, $\mathcal{K}$, *possibly with subscripts, denote some keyword query, concept, and keyword resp., then* $\mathcal{Q}$ *is a set of keywords* $\{\mathcal{K}_1, \ldots, \mathcal{K}_i | i \geq 1\}$, *whose semantics is denoted by the set of concepts* $\{\mathcal{C}_1, \ldots, \mathcal{C}_j | j \geq 1\}$, *for some positive integers* i, j. Query Expansion *is a query reformulation technique that appends to* $\mathcal{Q}$ *a (possibly empty) set of keywords* $\{\mathcal{K}_m, \ldots, \mathcal{K}_{m+n}\}$ *while retaining the semantics of* $\mathcal{Q}$, *for some positive integers* m, n.

Definition 1 ensures that there exists *at least one* concept for every non-empty keyword query. In addition, the number of concepts may not equal that of keywords. Query expansion does *not* mean to expand concepts presumed in a query (i.e., the query intent remains intact) but to augment the keyword set by including terms more relevant to the concepts such that the query intent becomes more concrete and "tangible" to search engines. According to definition 1, two special cases might occur. When all the keywords relate to one concept, this QE coincides with concept-based query expansion shown in [Qiu and Frei, 1993]. When every keyword corresponds to an individual concept, the keywords may be considered independent of each other.

Definition 2 (Entity). *An* entity e *is defined by*

$e \quad := \quad e^a \quad | \quad \{e_1, \ldots, e_n\}$,

where an atomic entity e^a *refers to a piece of information unit considered indecomposable under some context.*

The actual interpretation of entities depends on the context. For example, the name of a person may be considered an atomic entity in some context, but it may turn out an entity composed of two atomic entities $\{first\_name, last\_name\}$ in another context.

Definition 3 (Document). *A* document $\mathcal{D}$ *is a collection of entities* $\{e_1, \ldots, e_i\}$ *for some positive integer* i. *A* domain *is a section of the world expressing some specialized* subject knowledge. *Such knowledge spreads over a* domain corpus $\mathbb{D}$, *i.e., a collection of documents* $\mathbb{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_k\}$ *for some positive integer* k.

A document itself is *not* an entity but contains a rich set of entities. In a domain corpus, an entity can occur in any document as many times as possible.

2.2 The Effects of Domain Knowledge

The application of background knowledge is substantiated by the assumption that subject knowledge does have positive effects on query expansion. This assumption has already been validated, e.g., [Hsieh-Yee, 1993; Sihvonen and Vakkari, 2004]. Interestingly, the way the users behave when employing domain knowledge depends on variables like the level of task difficulty and their expertise in the domain.

Expert or experienced users, i.e., users that are familiar with the topics involved in queries, were found to call for more relevant terms in formulating queries by exploiting their own domain knowledge, i.e., using their own terms [Hsieh-Yee, 1993; Sihvonen and Vakkari, 2004]. When they were tested with queries outside their areas of expertise, expert users extensively took advantage of knowledge sources for term suggestion [Shute and Smith, 1993]. It seemed that novice users, as opposed to expert users, rarely consulted the thesaurus or other knowledge sources [Hsieh-Yee, 1993; Sihvonen and Vakkari, 2004]. The appreciation of domain knowledge is limited for novice users. However, it is interesting to observe that novice users still uses their limited knowledge of the

field spontaneously to generate terms [Shute and Smith, 1993], although a low level of knowledge requires more changes of the initial query (thus more terms are used) [Wildemuth, 2004].

To sum, domain knowledge affects user behavior in that high domain knowledge lead to more efficient term selection strategies and less errors in search tactics [Wildemuth, 2004]. Furthermore, it seemed that terms selected by experts tend to gather more around the query concepts [Sihvonen and Vakkari, 2004] and that experts tended to formulate shorter queries [Duggan and Payne, 2008]. The application of domain knowledge help users reduce noise in obtaining more useful terms. Altogether, It is also plausible to state that users' familiarity with the search domain has a great effect on retrieval results. Undoubtedly, an effective *semantic* query expansion strategy is expected to leverage the domain knowledge.

2.3 The Effects of Ontologies

The performance of information retrieval can be improved either by making the queries more comprehensible to the documents or vice versa. The difficulties lie in the way that query terms relate to documents. A substantial body of research on ontology-guided QE, as early as [Biswas et al., 1986], reveals that ontologies may bridge the gap between query terms and documents through semantic mechanisms. Specifically, adding ontology to QE approaches was described concisely as having the consequences of "an increase in the effectiveness of retrieval and a decrease in the efficiency of text processing" in [Croft, 1986].

A thesaurus in this paper is considered a simplified ontology, for it lacks a formalism for representing the domain semantics [Nagypál, 2005]. Thesauri/ontologies may be used solely on the query side as a source of relevant terms, however, they may also be involved in query processing (the actual retrieval). Typically, in the latter case, ontologies may be exploited to disambiguate queries, annotate or index documents, compute similarity between queries and documents, and so on, as shown in [Biswas et al., 1987]. It appears that using ontologies on the query side alone is insufficient to improve retrieval effectiveness, but this claim is not fully substantiated in the literature. This paper highlights the discussion about this problem.

Ontologies An ontology is a Knowledge Base (KB) expressing the subject knowledge of a domain. In this paper description logics underly the ontologies as the knowledge representation protocol. A domain ontology is characterized as a combination of *intensional knowledge* and *extensional knowledge* of its domain. Specifically, intensional knowledge, similar to a database schema, describes the structure of knowledge in the domain (referred to as a TBox); extensional knowledge expresses instances in the domain (referred to as an ABox), similar to tuples (data) in databases. Unlike databases, the distinction between schema and data in ontologies may be blurred by introducing the concept construct *nominal*. Every ontology has a set of vocabulary descriptions, called its *signature*, that consists of concepts ($\{C\}$), roles ($\{R\}$) and instances ($\{I\}$).

A concept C is defined inductively, for instance in the basic description logic $\mathcal{ALC}$, as: $C, D := A|C \sqcap D|C \sqcup D|\exists R.C|\forall R.C|\neg C$, where A is an atomic concept and R an atomic binary role. The expressiveness of this logic can be extended by introducing more concept constructs. Semantics and other details are further explained in [Baader et al., 2003].

Intensional knowledge about a domain is expressed as a set of axioms in a TBox, in the form of $C \sqsubseteq D$, stating that C is a *subconcept* of D. Conventionally, $C \equiv D$ is written as a shorthand for $C \sqsubseteq D$ and $D \sqsubseteq C$, meaning that C is logically *equivalent* to D.

Extensional knowledge exists in an ABox that is shown as assertions about instances. A *role assertion* of the form $R(I_1, I_2)$ specifies the relationship between the two instances, and a *concept assertion* in the form of $C(I_1)$ indicates the concept that the instance belongs to.

3 A Synthesized View

There were different systems for applying domain knowledge to query expansion. Most of the systems share the same strategies, consequently, this section unifies these ontology-based QE approaches to allow for easy implementation and comparison across systems.

3.1 An Overview

The synthesized plan, depicted in Fig. 1, consists of three major components. The preprocessor, accepting raw user queries as the input, analyzes the queries and applies some query cleaning or disambiguation techniques to them. Both syntactic and semantic techniques may be employed in the preprocessor to obtain a "clean" copy of the raw query, possibly with annotations. The preprocessor is briefly discussed in Sect. 3.2.

The core component is query expansion, achieved in several phases. Before any actual query expansion, the mapping between the vocabulary of the ontologies and that of the corpus is to be built and maintained. The mapping is indispensable for retrieval improvement using ontology-based QE approaches (Sect. 3.3). The mapping in principle can be obtained in two ways. On the one hand, the corpus to be queried against may be annotated (Sect. 4), with the help of the domain ontologies. This process explains the label "Annotation" in Fig. 1. On the other hand, a corpus-relevant ontology can be constructed from the corpus (Sect. 5), i.e., the label "Construction" in Fig. 1. A query expansion algorithm that focuses on term selection is executed to solicit candidate terms from multiple sources (Sect. 6).

A post-processor (Sect. 7) serves the purpose of optimizing the set of candidate terms output by the query expansion component and of formulating new queries to be evaluated.

3.2 Query Annotation

Whether a user query fits in the domain underlying an ontology is a fundamental question to ontology-based QE. This paper assumes that the submitted user queries are within a particular domain (or against a domain-specific corpus) so that the domain ontologies can be directly utilized. Otherwise, the domain of a query has to be determined in the query annotation phase, e.g., leveraging the user profiles [Dumais et al., 2003], feedback-style feature (concept) selection [Egozi et al., 2008], manual assignment (user selection) and so on.

A mapping between queries and ontological entities seems necessary. Bhogal et al. [2007] attributed the mismatch between query terms and ontological concepts to ontology design, which ignores the fact that a query can take on various forms to express the same information needs. User queries are ambiguous because they are too short to express the user information needs precisely, in addition to the inherent ambiguity in natural languages. Details about query disambiguation techniques can be found in the literature, e.g., [Pu and Yu, 2008; Stojanovic, 2005].

Traditionally, query keywords are assigned a weight to probabilistically discerning relevant from irrelevant documents. Furthermore in most IR systems stop words, whose weights are negligibly small, are stripped off the keyword queries. We assume hereafter that the keywords are well-formed,

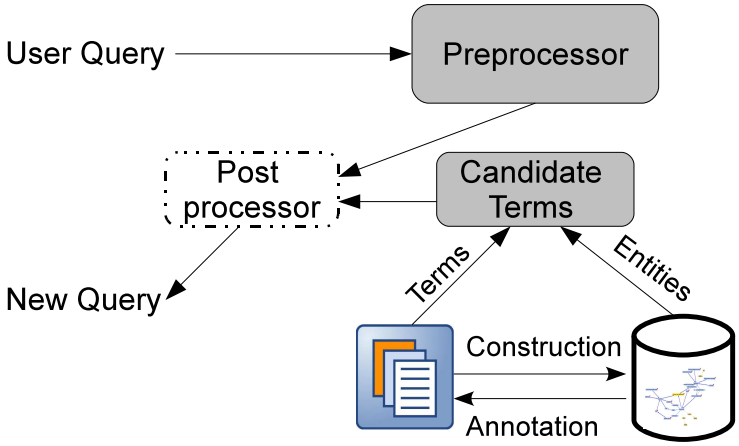

Fig. 1. A Synthesized Plan for Ontology-based QE

i.e., they adhere to predefined syntax. This assumption rules out the possibility of syntactic problems in queries.

Queries can be preprocessed to identify *important* keywords that can be linked to entities in ontologies or corpora to help identifying query intent, e.g., [Kim and Kim, 1990; Dey et al., 2005]. Typically, query terms are annotated in the same manner as corpora are, for example [Croft, 1986; Srinivasan, 1996c; Aronson et al., 1994; Aronson and Rindflesch, 1997; Crouch and Yang, 1992]. Compared to a corpus, queries may be subject to more thorough annotation due to its smaller size. A comprehensive analysis of a query may tokenize the word and sentence boundaries, tag the words for their part-of-speech, segment sentences to recognize noun and verb phrases, eliminate stopwords and so on [Moldovan and Mihalcea, 2000]. To resolve polysemy of words, query disambiguation techniques can be used. For instance, Moldovan and Mihalcea [2000] ranked the word senses based on the number of hits of web searches that are composed of one sense of the word, the corresponding synonyms specified in WordNet for that sense, and the original other words. The rank order of senses was further refined in [Moldovan and Mihalcea, 2000] by introducing the conceptual density between two words. That is, the method measures the density of common concepts between the hierarchies of the two words in their WordNet glosses, which have the explanatory comments and examples on concepts. Some studies instead assign ontological concepts to queries manually, as shown in [Voorhees, 1994; Hersh et al., 2000; Wollersheim and Rahayu, 2005b; Dey et al., 2005], or conversely, the keywords can be manually linked to ontological concepts [Järvelin et al., 2001]. In an interactive environment, e.g., [Tudhope et al., 2006], users are prompted to select ontological concepts for keywords.

We believe query annotation, which interprets a query as concepts, is particularly indispensable for concept based query expansion. However, ambiguity is inherent in natural languages, hence it may not be completely resolved by automated procedures, sometimes even impossible by human beings. Annotation does not guarantee that the queries are free from ambiguity. Disambiguation may be partly resolved either syntactically or semantically, using approaches presented in, e.g., [Pu and Yu, 2008; Pound et al., 2010]. Query ambiguity may be further relaxed by heuristics. For instance, Calegari and Pasi [2008] attempts to personalize query expansion using the Google

desktop widget. Similarly, mining user behaviors may as well be helpful, as shown in [Dumais et al., 2003]. Cui et al. [2002] made use of query logs to reflect the preferences of a majority of users, which decreases the probability of presenting irrelevant information to users. Fu et al. [2005] maintained a log file of statistical data for interpreting fuzzy spatial relationships like *near*. Also as discussed in Sect. 4.1, facets are handy for understanding query intent as users normally have a vague idea about what they are looking for.

3.3 Linking Corpus and Ontology

Definition 1 raises some important issues. First of all, it is difficult to discover the query intent from user queries and have the query intent represented in a combination of concepts. In addition, given ontological concepts, significant efforts may be required to build the association between keywords and concepts. For query expansion, it also remains to study how to appropriately select additional terms that are considered relevant to the query intent. Researching possible solutions to these issues is the central theme of this paper. There is a lack of research in terms of how to effectively and efficiently relate the domain corpus to the ontology. This paper first state why an ontology should reference the content of documents in a corpus. As a consequence, the paper elaborates how can an ontology "understand" a corpus.

Ontologies' Dependency on Domain Knowledge Ontologies may well capture domain knowledge if the ontology engineers have a solid understanding of the domain, because the quality of domain ontologies relies heavily on a "correct" representation of the domain knowledge, either by way of manual construction or by some automated tools. Evidences were shown in existing systems. In [Voorhees, 1994], Wordnet, a hand-crafted thesaurus independent of any particular domain, provided very limited improvement on query expansion. Contrarily, using thesauri built from a corpus, e.g., [Qiu and Frei, 1993], improved retrieval effectiveness moderately. Jones et al. [1995] also suggested that highly structured and rich thesauri designed for a particular domain should be built, when they witnessed no improvement on retrieval performance using thesaurus-based query expansion. Various reasons account for the fruitless attempts of using corpus-independent ontologies in query expansion, one of which is that these ontologies are too general to capture specific domain facts, for instance, proper nouns included in a query may be overlooked.

As stated in Sect. 2.3, empirical evidences showed that using corpus independent thesaurus (ontologies) query expansion "does not necessarily improve searching performance" [Hersh et al., 2000], as opposed to other expansion strategies based on retrieval relevance [Srinivasan, 1996c] and term co-occurrence analysis [Buckley et al., 1994b]. In [Hersh et al., 2000], UMLS was used to expand queries in MEDLINE annotated by the MeSH ontology. Because MeSH terms in MEDLINE already lead to non-trivial retrieval improvement [Hersh et al., 1994], it might be the case that ontology-based query expansion has limited capability to further improve the result. Nevertheless, results shown in [Hersh et al., 1994] suggest that a corpus-dependent ontology is more effective, and [Aronson and Rindflesch, 1997] empirically showed that ontology-based query expansion is effective in some cases only when documents are MeSH-indexed. More solid studies for this question [Srinivasan, 1996c; Aronson et al., 1994] showed that the mapping from the corpus to the metathesaurus/ontologies is crucial. Yang and Chute [1993] thus proposed the linear least square fit mapping for such a purpose, which automatically learns from a training set of relevant queries and documents. Similarly, Yang [1994] introduced an expert network to generate a mapping between training queries and training documents.

Therefore, we conclude that an ontology should be corpus-aware to significantly improve retrieval efficacy. Analogous to [Xu and Croft, 2000; Bai et al., 2007], ontologies, representing domain or general knowledge, provide a *global* context, while the corpora being queries provide a *local* context. Consequently, building the correspondence between the ontologies and the corpora amounts to combining the global and local context analyses. The remaining question is on the granularity of mutual understanding between an ontology and a domain corpus. A mapping to promote such understanding tends to be expensive to build. Hence, our aim is to minimize the efforts in annotating the corpus while maximizing the performance gains. Mapping can be constructed in two major ways. One is to tag the corpus with the ontological entities, typically concepts. The other is to augment the ontology by drawing missing salient terms from the corpus. The first approach is discussed in Sect. 4, and the second in Sect. 5.

Ambiguity may exist in the mapping. In the first place, words in the ontologies are themselves ambiguous[6], and their participation in annotating a corpus may lead the ontology-corpus mapping astray. Other than that, ontological entities mapped to ambiguous words in the corpus also present ambiguity, as briefly discussed in Sect. 4.3. Under such occasions, disambiguation seems important for ontology-based approaches to outperform the others. Though this problem is open, we still witness attempts to resolve ambiguity using contextual information, e.g., Aronson et al. [1994]; Rindflesch and Aronson [1994] partially disambiguated words based on the distribution patterns of semantic types in text.

4 Corpus Annotation

A corpus may be structured in a way friendly to human beings, however, it can only be vaguely understood by computers. Corpus annotation provides a mechanism for computers to partly make sense of documents in the corpus, generally by annotating important document entities with tags that are friendly to computers. Such tags, sometimes called meta-data, markups, etc., may be provided a prior or derived from additional sources, e.g., ontologies. Nevertheless, notice that the ontology contributing to the tags may not necessarily be the one used for query expansion though in general the two ontologies share a common vocabulary for retrieval effectiveness. For example, in [Hersh et al., 2000] MeSH metathesaurus is used to annotate MEDLINE collections, while UMLS is employed to query against MEDLINE. Corpus annotation enables a correspondence between the documents and the ontological entities.

In practice, some popular domain corpora, including MEDLINE and CDA documents, directly refer document terms to ontological entities, hence requiring no corpus annotation [Rada and Bicknell, 1989; Tudhope et al., 2006; Farfan et al., 2009; Liu et al., 2009]. In particular, Rada and Bicknell [1989] even represented every document as a set of ontological entities. We also noticed that some works expand the query solely based on the query itself and the ontology, completely avoiding annotating the domain corpus [Navigli and Velardi, 2003].

When a corpus needs to be annotated, either a manual or an automatic process surface. Manual annotation is common for small corpora, where documents can be semantically marked up by human experts [Biswas et al., 1986; Aronson and Rindflesch, 1997]. Automatic corpus annotation are challenging, yet several tools are available, e.g., [Dill et al., 2003; Kiryakov et al., 2004], aiming to enrich textual web content with metadata. For instance, [Dill et al., 2003] used the TAP ontology

[6] Concepts in ontologies, in general, are precisely defined and free from ambiguity, but the words/terms in their definitions are not.

[Guha and McCool, 2003], an evolving general-purpose ontology, to resolve ambiguity; MetaMap [Aronson, 2001] is used in [Wollersheim and Rahayu, 2005b] for mapping biomedical text to the UMLS metathesaurus and extracting concepts from Ohsumed documents. Because of scalability and ontology availability, automatic annotation may be unavailable in some cases. In particular, the Web, which represents the most general and complicated domain, poses great challenges for any search engines that try to "understand" its content by way of annotation. The focus of this paper is to annotate domain corpora, which are significantly smaller than the web corpus.

4.1 Annotation Granularity

Document annotation can be accomplished in a variety of ways because of the number of structuring units that can be considered for annotation, i.e., the granularity of annotation. It would be a gargantuan undertaking for a large corpus to be annotated in the finest grain. Therefore, the granularity of annotation has a direct impact on the success of linking corpora entities to ontological entities and on the efforts devoted in annotation. The following discussion presents several types of annotation, from the most granular to the most abstract annotation.

Words Intuitively, word level annotation as shown in [Buckley et al., 1994b; Hersh and Hickam, 1995; Müller et al., 2004; Díaz-Galiano et al., 2009; Castells et al., 2007] provides the finest granularity for recognizing the basic units. This is mostly achieved by tokenizing the documents, performing linguistic stemming, providing part-of-speech (POS) tags, importing additional tags (e.g., bibliographic tags) and so on. As the most widespread way to annotate a corpus, full word-based annotation enables retrieval algorithms to access every composing units in the corpus, however, it requires heavy, costly preprocessing on large volume of data. Word level annotation gives rise to a collection in shortage of semantic information, an big disadvantage for it being used in semantics-oriented applications. Term-based annotation, where every document is characterized as a set of pairs (τ_i, w_i), specifies a term occurring in the document with its association degree, e.g., [Kim and Kim, 1990]. Because of its computation intensity and syntactic nature, term-based (bag-of-words) annotation is considered on the word level. Detailed discussion on word-based annotation can be found in other literature, e.g., [Manning et al., 2008].

Named Entities Expert users, instead of entering topical queries, opt for more objective information pieces, e.g., names of people, organizations, and so on, in formulating their queries [Markey, 2007; Dalvi et al., 2009]. Such information pieces correspond to the named entities, a subset of the entities defined in Def. 2. The usual notion of *entities* in the literature, e.g., [DeRose et al., 2007; Cheng et al., 2007], is more suitable for *named entities* in this paper. Consequently, a coarser level of annotation, compared to word-level annotation, may only identify certain kinds of entities in documents. In some systems, e.g., [Dumais et al., 2003], entities were especially prevalent in query logs, hence the extraction of named entities like people, locations, etc. are essential for information retrieval. Such systems may only identify named entities from within the boundaries of specific document entities like noun phrases, e.g., [Bast et al., 2007]. The main advantage of such a restriction is that the granularity of entity annotation is easy to manage. Corpus of certain types of documents may favor particular kinds of entities, for example, news articles are mainly concerned with named entities in the stories, which can be automatically recognized by tools as in [Dakka and Ipeirotis, 2008]. Castells et al. [2007] required additional document properties (authors etc.) be extracted. The types of named entities to be extracted rely on the corpus and the aimed tasks.

Jing and Croft [1994] showed that among different types of words nouns contribute most to improving retrieval results. The most popular way for named entity recognition is based on noun phrases, a kind of collocations [Manning and Schütze, 1999, Chap. 5]. For instance, [Aronson, 2001; Jing and Croft, 1994; Aronson et al., 1994; Wollersheim and Rahayu, 2005b] split text fields into noun phrases for identification and normalization. In [DeRose et al., 2007; Müller et al., 2004] a lexicon of commonly seen names of interest is manually maintained for recognizing entities. Bast et al. [2007] computed for each word or phrase the most *probable* entities, based on [Dill et al., 2003]. Noun phrase identification on the word level is very costly for large collections, but it may be relatively easier for some data sources like DBLP and medical data collections, where a document is normally organized by fields [Wollersheim and Rahayu, 2005b; Nadkarni et al., 2001].

Concepts Concepts are more abstract than named entities, though named entities may also be counted as concepts, e.g., the named entity *people* is a suitable concept. In the literature, the term *concept* may be used interchangeably with *named entities* [Dalvi et al., 2009]. In general, named entities are primarily concerned with *instances*, e.g., an individual person, while concepts describes a class of such instances.

Concept recognition, more challenging than noun phrase recognition, is beneficial to ontology-based query expansion because of the rich semantics with concepts. There are different approaches for concept identification in documents. By extending noun phrase identification, simple concepts in phrases can be extracted, e.g., [Aronson, 2001]. The downside of this simple extension is that some contextual information may be missing due to the small units of text used, e.g., noun phrases. Alternatively, larger units of text like sentences can be used [Hersh and Hickam, 1995; Hersh et al., 1992], but this approach, being more computationally intensive, tends to produce more false positives than the phrase-based one, because more concepts clutter in a larger unit of text, e.g., a sentence, such that concepts that would not have been implied in a smaller context may be generated [Nadkarni et al., 2001]. Identification of concepts within documents can also be realized by other techniques, e.g., explicit semantic analysis in [Egozi et al., 2008], rule-oriented approaches [DeRose et al., 2007], etc. Nevertheless, several challenges exist for these techniques, for example, the concepts generated are noisy and ambiguous. For instance, Brauer et al. [2010] reported that 98% of the matches were ambiguous. To handle such problems, assigning concepts to documents manually was used in [Biswas et al., 1987], while Brauer et al. [2010] selected the longest concept that matches the textual entities to reduce ambiguity.

Document Feastures The least detailed annotation that analyzes a document as a whole correspond to document classification, i.e., generation of facets for documents. Such annotation are feature based. For instance, Grootjen and van der Weide [2006] used an automatic indexer to assign attributes (words or short expressions) to every document.

Document features[7] are a special kind of concepts. Such features, normally as mutually exclusive document descriptors, provide more contextual information for query expansion. Particularly, interactive systems may prompt users to choose a subject for query expansion [Greenberg, 2001b; Müller et al., 2004; Navigli and Velardi, 2003] based on other sources (e.g., using ProQuest Thesaurus in [Greenberg, 2001b]); non-interactive systems can classify documents to generate document features [Yang, 1994; Wollersheim and Rahayu, 2005a; Crouch and Yang, 1992]. An example is Chang et al.

[7] Document features are document descriptors, which are not to be confused with features/attributes that depict properties of objects. The term *facet* is used as a synonym for *document features* in this paper.

[2006], where salient concepts, generated from selected documents by feature extraction, are considered the main topics of those documents to further reformulate the user queries.

Some corpora, e.g., bibliographics datasets like DBLP, have built-in facets like authors, titles and so on. It is also common to extract features from documents, e.g., Biswas et al. [1987] relied on human experts to divide the document space into several facets. Castells et al. [2007] required that every ontology have a facet as one of the main base classes to annotate documents. The recognized facets come from the documents or ontological concepts, which can form taxonomy for query expansion [Dakka and Ipeirotis, 2008]. An example given in [Castells et al., 2007] is that the term *Irises* is more relevant to the concept *Painting* for Van Gogh's work instead of *Flower*, given the word appears in a document under the topic of *Art*. Therefore, facets reduce term ambiguity by narrowing down searches to a more restricted context. The readers can find more approaches on recognizing facets in the literature, e.g., [Chakrabarti et al., 1998].

4.2 Corpus Indexing

Corpus indexing, specifically used as a reference to document entities, is an implementation of the mapping between ontologies and corpora. Indices, which may be consulted in query annotation to identify the possible query concepts, are largely utilized in the process of term selection. It is thus important to have well-built indices to achieve better performance. There are two fundamental aspects of corpus indexing, one of which is *indexing exhaustivity* that measures the number of terms collected in indices, the other being *specificity* on the level of indexing detail.

Studies on indexing specificity showed controversial results, as there is no agreement on what is the *optimal* way to index document contents: words, concepts or both [Hersh et al., 1992]. It appeared that word-based automated indexing dominates the indexing approaches, nevertheless, other studies, for example Aronson et al. [1994]; Yang and Chute [1993], supported the use of ontologies for concept indexing. Nadkarni et al. [2001] argued that concept indexing takes advantage of semantics of text, but accuracy remains a problem, which restrains the use of concept indexing on large collections of documents. After all, Srinivasan [1996a] empirically demonstrated that the most successful document-indexing strategy is to combine document indices and concept indices for retrieval. Despite all the controversies, indexing documents for IR is mostly done on the syntactic level, e.g., by statistically indexing on words. Though a few systems index concepts appearing in documents, e.g., Hersh and Hickam [1995]; Nadkarni et al. [2001], word-based approaches have been prevalent. In [Voorhees, 1994; Srinivasan, 1996c; Järvelin et al., 2001; Srinivasan, 1996b; Farfan et al., 2009] the documents are fully indexed on words. Farfan et al. [2009] also computed the *tf-idf* score, and Srinivasan [1996c,b] further parameterized indices for term frequency, inverse document frequency and the length of a document to test the impact factors on retrieval effects of different indexing strategies. Srinivasan [1996b] showed that indexing strategies without inverse document frequency decreases performance compared to the others. Interestingly, we observed that a synthesis of word-level indexing and concept-level indexing has been reported as well, e.g., [Bai et al., 2007], where the TREC-8 test collection was indexed in words and in concepts and less improvement was gained for concept-based retrieval.

Indexing exhaustivity is less exposed to questions. As long as the indices are not too large to cause performance issues, the more terms used in indices the better. Therefore, a full-text indexing is satisfactory for collections in small size. If entities in a corpus are already identified, Kiryakov et al. [2004] suggested that indexing documents in the corpus with respect to entities allows more accurate searches for queries that place constraints on entities.

4.3 Applying Semantic Markups

When mapped to an ontology, words or terms in the text are annotated with semantic mark-ups (i.e., tags) from the ontology. The difficulty lies in how to accurately attach tags to entities in documents. Fully automatic process of semantic annotation is very challenging, although [Castells et al., 2007; Kiryakov et al., 2004] provide a scheme for semi-automatic annotation to link ontological terms to texts in documents. For instance, Kiryakov et al. [2004], during information extraction, mapped each entity reference in the text onto the specific instance together with the most specific concepts in ontologies.

The annotation process chronologically may take several steps including variant generation, ontological candidates identification and matching, depending on the accuracy requirement.

Variant Generation To generate variants of simple noun phrases, other lexicons or knowledge bases may be used to initially identify synonyms, abbreviations, acronyms and so on. Variant generators also take advantage of predefined morphological rules for computing inflections, spelling variants and others [Buckley et al., 1994b; Aronson et al., 1994; Aronson, 2001; Croft, 1986]. In [Müller et al., 2004], regular expressions were used as a substitute for rules to obtain terms of variable forms. To keep track of the generation of variants, a distance value capturing the *adjacency* information can be computed between every variant and its history phrase to generate similar variants.

Mapping Candidates An ontological candidate must meet certain conditions, e.g., it contains at least one of the variants [Aronson, 2001; Aronson et al., 1994]. To quantify this evaluation, a similarity value between a text phrase and a candidate can be computed in terms of a number of factors, e.g., candidate coverage, distance values, and so on [Aronson, 2001], in order to measure how much of a candidate matches the original term. A list of ranked candidates can then be produced and used for constructing the mapping. Observe that ambiguity may arise during this process. Exploiting contextual information for the purpose of word sense disambiguation can reduce the number of incorrect mappings [Rindflesch and Aronson, 1994], but the solution to this linguistics problem is still open.

4.4 Corpus Structure

The data format of the corpus in question has a direct impact on ontology based query expansion efficacy and the retrieval results. The data models used to organize information in a domain vary, and can be roughly categorized in terms of the *tightness* of relationship between entities. In this paper, we distinguish different data models by identifying the *features/attributes* extracted for representing documents. Fully structured data, in particular database relations, is a collection of values of features defined in the schema. Structured data is easy to manipulate by machines, but incomprehensible to humans. This paper focuses on semi-structured and unstructured data.

Unstructured Data Compared to data that strictly conforms to some data models like relations in databases, unstructured data is not constrained by any structures but is simply free and human-friendly text. The prominent unstructured data set is the web, which is distinct from other datasets because of its size and unreliability of some data. Being structure-free, there exist no predefined features or values in documents at all, and proses in the documents are difficult to parse or understand even by humans due to various linguistic reasons. In order to use techniques like query expansion, annotating the documents to extract certain features is indispensable. For instance, the TREC collection consists of english prose from various sources.

Semi-structured Data Semi-structured data does not follow specific formal structures but is annotated with tags to highlight certain information in it, e.g., hierarchies of records, data fields and so on. It is loosely structured to enable machine-readable, object-oriented information, where a limited set of features exists to allow for easy access to some parts of the data. Without loss of generality, features can be interchangeably used with tags here.

XML is a prominent example, where an XML document is composed of its content and markups. For example, in the medical domain Clinical Document Architecture (CDA)[8] leverages the use of XML. For keyword searches on XML documents, the keywords are matched to XML nodes that are covered by a minimal tree to be returned to users. To exploit the rich annotations provided by XML documents, ontological knowledge are employed to support domain-specific search, e.g. [Theobald, 2003; Farfan et al., 2009].

Bibliographic records are another semi-structured data example, where the document number, title, authors and so on are typically tagged for every document in the collection, as can be seen in the DBLP collection of data[9]. Particularly, there exist a large number of works that address ontology-aided search on the MEDLINE collection, e.g., [Rada and Bicknell, 1989; Srinivasan, 1996c,b].

5 On Domain Ontologies

Several ontology-aided query expansion systems [Kasneci et al., 2008; Castells et al., 2007; Bast et al., 2007; Theobald et al., 2008; Dey et al., 2005; Vallet et al., 2005] assumed the availability of full-fledged ontologies, as automatic construction of ontologies shows great difficulties due to several factors, e.g., the inability of extracting taxonomy precisely as stated in [Cafarella et al., 2007]. However, semi-automatic and manual creation of ontologies are feasible in many cases.

A domain ontology can be crafted regardless of the underlying corpus that the queries are searched over. Otherwise, an ontology can be drawn from a particular corpus to enhance searching in that corpus. Naturally, corpus independent ontologies are unaware of the corpus content, which are more suitable for "static" document collections. Otherwise, corpus-dependent ontologies are indeed necessary for documents that are frequently updated, because the vocabulary in the ontologies are subject to changes as well. Yet, building ontologies dynamically for corpora under constant updates, e.g., the Web documents, is challenging and costly.

It is the vocabulary used in a corpus and an ontology that matter, as discussed in Sect. 3.3. Recall that traditional thesauri, as a form of controlled language, select a list of collection terms from a corpus to provide a mediating interface between the documents and users. This section thus addresses how to build the vocabulary of an ontology, the mediating interface, out of a corpus. We explore some techniques in building ontologies from the underlying corpora. These techniques are roughly categorized by the granularity of extraction, namely, the type and quantity of entities to be chosen for the ontologies.

5.1 Corpus Independent Ontologies

We start our exploration with full-fledged ontologies independent of any corpora. WordNet has been widely used in query expansion tasks, for example, Voorhees [1994] expands a query with words that are lexically relevant to some keyword via WordNet. YAGO [Suchanek et al., 2007], a large

[8] http://www.hl7.org/implement/standards/cda.cfm
[9] www.informatik.uni-trier.de/~ley/db/

ontology built from Wikipedia pages and WordNet, serves as a domain-specific ontology as well as a general ontology. Another large general purpose ontology is TAP used in [Guha and McCool, 2003].

There are mature ontologies designed by domain experts in some domains as well. Specifically, well-founded biomedical ontologies emerge rapidly [Bodenreider, 2008]. For instance, [Rada and Bicknell, 1989; Díaz-Galiano et al., 2009; Srinivasan, 1996c,b] used the Medical Subject Headings (MeSH) thesaurus[10]; [Hersh et al., 2000; Aronson and Rindflesch, 1997; Wollersheim and Rahayu, 2005a] exploited the UMLS metathesaurus[11]. SNOMED CT[12], considered "the most comprehensive, multilingual clinical healthcare terminology in the world", was referenced for ontological search in [Farfan et al., 2009]; GO (the Gene Ontology[13]) focusing on gene and gene products is the core source for the ontology used in [Müller et al., 2004]. In addition to medical ontologies, LOIS, exploited in [Schweighofer and Geist, 2007] for query expansion, is a set of multilingual lexical ontologies for the legal domain. Likewise, Tuominen et al. [2009] utilized the library of, either specialized or general, ONKI ontologies. In these domains, the domain vocabulary is well shared and receives consensus in the community. Therefore, ontologies built on the vocabulary can be fully exploited by users. Nevertheless, well-founded ontologies are inaccessible in many application domains. The advantage of using a corpus-independent ontology is evident: it avoids domain corpus analysis and saves users tremendous workload in ontology engineering. Despite that, employing a general-purpose ontology suffers from the inability of capturing specialized knowledge since such an ontology may overlook concepts like proper nouns and highly technical terms [Voorhees, 1994].

5.2 Corpus Dependent Ontologies

While some ontology based expansion approaches just assume the availability of domain ontologies, there were attempts to construct corpus-dependent ontologies automatically (e.g., [Liu et al., 2009]) or through interaction with domain experts or users [Croft et al., 1989; Lee et al., 2008; Greenberg, 2001a].

Most existing approaches to automatic thesauri construction are based on the statistical co-occurrence of words in the collection, which is difficult to measure low frequency terms (the terms with very low *document frequency*). According to Zipf's Law, infrequently occurred terms are, however, good discriminators for clustering documents into thesauri classes that can be used for indexing [Crouch, 1990]. Note that these approaches to build thesauri are syntactic. A major disadvantage of the statistical approaches is that thesauri that are dependent on the document collection may need to be recomputed as the document collection changes.

Ontologies, on the contrary, take advantage of domain knowledge on a semantic level. While terms are congregated in ontologies, they are immune to corpora change, as they are semantically motivated and keep consistent meaning in different context. Qiu and Frei [1993] used a different way to build a similarity thesaurus statistically, where the meaning of a term is represented by a document vector space, in contrast to the traditional term vector space. Although retrieval effectivenss has been witnessed on small collections, the semantics of the domain knowledge is only

[10] MeSH is a comprehensive controlled vocabulary used for indexing life science articles, like the MEDLINE/PubMed database, and it also serves as an ontology.

[11] http://www.nlm.nih.gov/research/umls/. Note that the metathesaurus of UMLS combines information from various sources, the dominant one being MeSH.

[12] http://www.ihtsdo.org/snomed-ct/

[13] http://www.geneontology.org/

vaguely signified by a probabilistic QE model. Furthermore, (manual) construction of relational thesauri has been studied in Wang et al. [1985], and enhanced performance over statistical thesauri, since relational thesauri actually capture certain semantics of the corpora. From observations in [Brewster and Wilks, 2004] we can also see that automatic construction of ontologies (relational thesauri) is far more challenging than that of syntactic thesauri.

The Source for Authoring an Ontology Jing and Croft [1994] built an association thesaurus for query expansion, and suggested that larger collections yield better association thesaurus performance. The problem is the inefficiency of generating such thesaurus based on all the documents for large collections. Hence, Jing and Croft [1994] instead employed a representative sample of its collection. In fact, Müller et al. [2004] showed that the abstracts of medical articles, compared to the full text of those articles, yield higher overall recall but lower precision for keyword search, since full text introduces more noises to produce false positives[14]. Therefore, it might be desirable to compromise the source corpus size to achieve efficiency and possibly lower degree of ambiguity. Mandala et al. [2000] also concluded that constructing ontologies from a sample collection still significantly improves search results in the whole collection. Chang et al. [2006] also discovered that the construction of primitive concepts from the whole corpus is time-consuming, and "was promising only for poorly performing queries". Instead, [Chang et al., 2006; Grootjen and van der Weide, 2006] generated concepts based on the set of retrieved documents. The disadvantages are, borne of local analysis techniques, that concepts have to be rebuilt for every new query and that the quality of the set of retrieved documents may be unreliable. In general, it remains to see how to find the best sample collection for ontology construction. Liu et al. [2009] construct an ontology from the corpus annotation and SNOMED CT, i.e., the fact assertions that connect the documents with SNOMED CT are extracted and reasoned against SNOMED CT to infer new knowledge.

Terms that do not appear in the domain corpus may also be needed in a domain ontology. External sources, such as existing mature ontologies, databases, thesaurus and search engines, are all good sources for domain-dependent ontologies [Dakka and Ipeirotis, 2008]. For example, YAGO provides various levels of domain knowledge to serve as a basis for building a domain-specifc ontology. In [Müller et al., 2004] GO ontological terms contribute around 80% to the lexicon in the Textpresso ontology. Additionally, the Textpresso ontology was partly populated using terms from other biological databases like PubMed. It should be noted that some general knowledge may be missing in a corpus-dependent ontology. While refining user queries depends on domain knowledge, it may be more effective to take general knowledge into consideration. This has been studied in [Mandala et al., 2000], where three different types (one general-purpose Wordnet, the other two being corpus-dependent with statistics on co-occurrence of terms and grammatical rules, respectively) of thesauri for query expansion were combined. The problem for most domains is that not many well-founded ontology may be available to serve as good sources.

An ideally expressive and informative domain ontology will contain the details of all concepts in the corpus of interest. However, authoring such ontologies amounts to performing a complete analysis on the corpus, often undesirable or infeasible. It is often worth considering the trade-off between the efforts in ontology construction and the efficacy of exploiting the ontologies. A more practical approach is to select only those *salient* terms from the corpus to populate the ontology, manually [Nagypál, 2005] or automatically [Grootjen and van der Weide, 2006]. There is considerable

[14] This may not happen in corpora where facts are expressed without complex structures, for example, bibliographic data.

research on identifying terms and associations between them, e.g., [Biemann, 2005; Brewster and Wilks, 2004].

Automatic ontology construction can in principle "reverse" the annotation methods discussed in Sect. 4 to extract terms from a corpus. For instance, terms in a noun phrase can be extracted as an ontological concept. Cafarella et al. [2007] exemplified such extraction mechanisms, where the Web documents are extracted for facts (denoted as triples), types, and synonyms. Observe that the facts and types correspond to role and concept assertions, respectively. However, no known automatic techniques are available for extracting inclusion or functional dependencies so far. Several search engines also provide automatic extraction services, e.g., Yahoo term extraction service[15].

5.3 Ontology Expressiveness

The quality of ontology was thought of as one of the most critical factors incluencing the retrieval performance [Kim and Kim, 1990]. An ontology, in terms of its quality, can be evaluated by the soundness (consistency), the representativeness of domain knowledge, reasoning effectiveness and so on. The ontological representativeness of domain knowledge is related to ontology expressivity, one of our main concerns in this paper. When an ontology is built, it is only an encapsulation of the domain knowledge as only some chosen *salient* document units will participate in the ontology. Furthermore, ontology development or maintenance tends to lag behind the knowledge evolvement. For these reasons, it may be necessary that domain knowledge be query-relevant. In [Croft, 1986] domain knowledge was acquired by system interaction with the users.

Though Navigli and Velardi [2003] suggested the importance of ontology expressivity, there is a trade-off between the expressiveness of ontologies and the improvement of retrieval results. Indeed, Kiryakov et al. [2004] argued that a light-weight ontology poor on axioms suffices for defining essential domain entities and allows for more scalable management. Remarkably, a great number of large-scale medical ontologies reside in the DL $\mathcal{EL}++$, which is sufficiently expressive to describe most domain facts while deciding core reasoning problems in polynomial time[16].

The exploration of ontology is centered upon relationships, i.e., the association between entities. The association corresponds to roles in the underlying DL terminology. It is therefore reasonable to interpret ontology expressivity in terms of relationships here. The core set of relationship shared by most existing works [Biswas et al., 1986; Wollersheim and Rahayu, 2005b; Navigli and Velardi, 2003; Croft, 1986; Järvelin et al., 2001; Wollersheim and Rahayu, 2005a; Farfan et al., 2009; Wang et al., 1985; Lee et al., 2008; Greenberg, 2001b; Fu et al., 2005; Greenberg, 2001a; Liu et al., 2009] can be characterized as follows.

Relationships of our interest may belong to the open or closed class [Green, 2001]. The closed class contains hierarchical relationships, e.g., the *IS-A* relations (hyperonymy and hyponymy), meronymy, holonym, synonym, antonymy and so on. New relationships are rarely created in the closed class, thus this class is enumerable. Most of relationships in the closed class receive general consensus and their semantics is explicitly defined. WordNet is a representative ontology that features the closed class of relationships: *IS-A* and *part-of* (meronym and holonym) relationships are heavily exploited [Croft et al., 1989; Kim and Kim, 1990], sometimes the *IS-A* relation being the dominant one [Voorhees, 1994]. Wang et al. [1985] also enumerated several groups of semantic relations for experiments. Nevertheless, the most effective relations are idiosyncratic to the particular query and the target text collection [Greenberg, 2001a].

[15] http://developer.yahoo.com/search/

[16] http://www.w3.org/2007/OWL/wiki/Standalone_Profile:_EL++

The open class of relationships typically involve associative relationships like *cite*, *related-to*, *see-also*, *provenance*[17] and other referencing relationships, typically seen in thesauri. The membership of this class can not be entirely enumerated, as new relationships of this type can be coined. The semantics of these relationships are normally undefined, relying on the the enumeration of their participating entities to imply the meaning. Features (or attribute-value pairs) also belong to the open class. For a particular corpus, this class of relationships may be more useful. For example, the relationship *cite* was assumed to represent the strongest plausible relationships between documents in [Croft et al., 1989]. Sihvonen and Vakkari [2004] demonstrated that related terms were productive expansion terms as well.

Different levels of retrieval improvement were observed in the literature, not due to the ontology expressiveness but because of the different expansion strategies; therefore, further research may focus on how to fully exploit the rich semantics in ontologies [Wollersheim and Rahayu, 2005b; Srinivasan, 1996c]. We believe that ontologies within the logic $\mathcal{EL}$++, where hierarchical relations, transitive (e.g., [Dolby et al., 2009]) and inverse roles are included, are sufficiently expressive to construct domain ontologies that most domain-specific query expansion strategies can apply to. Other concept constructs may be used, e.g., the relations characterized in [Wang et al., 1985] correspond to the DL $\mathcal{ALCI}$. It seems that cardinality restrictions on concepts (e.g., $\mathcal{N}, \mathcal{Q}$) and nominals ($\mathcal{O}$) seldom contribute to retrieval efficacy. In particular, most expansion strategies are most concerned with roles, therefore, we conjecture that the expressiveness of ontologies will most likely to benefit from adding more role constructors or role axioms instead of concept constructors.

6 Ontology-aided Query Expansion Algorithms

As mentioned in Sect. 5.3, how ontologies can be effectively exploited by an expansion algorithm remains a cursorily studied question. Previous sections elaborate the query and corpus annotation. In this section, we overview the state-of-the-art expansion algorithms to synthesize an algorithm that can be commonly employed.

We assume the query concept(s) exist and will be used during query expansion. The most straightforward way to expand a query is to traverse the ontological taxonomy along different relationships [Croft, 1986; Hersh et al., 2000; Wollersheim and Rahayu, 2005b,a]. The process can be roughly segmented into two phases, i.e., developing strategies to obtain sufficiently many candidate terms, discussed in Sect. 6.1, and designing ranking schemes to determine the most promising terms, discussed in Sect. 6.2.

6.1 Candidate Term Selection

The insufficiency of the traditional (syntactic) term selection strategies lies in that useful and useless terms can not be distinguished based on term distributions [Cao et al., 2008]. To distinguish good and bad terms in semantic approaches, we consider several criteria for term selection; therefore, this section focuses on what entities can be potential candidates.

[17] Provenance relationships express sources or origins. For instance, some *Wine* is related to *Merlot* by the provenance *madeFrom*.

Term Selection Tactics Sect. 2.2 indicate that information specialists take advantage of a variety of search strategies to obtain the desired information need. Presumably, an automatic query expansion algorithm is more effective if it captures the strategies employed by human experts. In this respect, rules for term selection, the core of query expansion algorithms, is set out in Table 1 to express the various tactics used by human experts w.r.t. the domain knowledge used.

Transformation	`Inflectional changes to terms, reorder the words in a term, etc.`
Hierarchy	`Broader or narrower terms`
Neighbor	`Synonym, related terms (either syntactically or semantically)`
Contrary	`Antinomy`
Provenance	`Additional terms from the source of already selected terms`

Table 1. Term Selection Tactics

The fact that human experts favor specific tactics for a particular term depending its the context indicates that the order and preference of applying the term selection rules are subject to change for different queries. Such human behavior is approximated by the underlying query expansion algorithm that select terms based on the prior information obtained from the queries.

Selecting Candidate Terms Candidate terms may be selected syntactically and/or semantically. A typical keyword search engine tends to match keywords to the terms in documents with the use of *syntactic* techniques based on term co-occurrence. What semantic selection strategies do is to draw terms from the sources (both documents and ontologies) *semantically*. A combination of syntactic and semantic approaches to obtain candidate terms is feasible, as shown in [Aronson and Rindflesch, 1997], as they are complementary, other than exclusive, to each other. In what follows we focus on the semantic selection strategies.

Ontological Entity Selection This section describes how candidate terms can be drawn from ontologies. Before an ontology is used in term selection, it must be processed by reasoners to make implicit knowledge available. Reasoning may be computationally expensive, yet tractable reasoning is possible for ontologies expressed in certain languages, e.g., ontologies in $\mathcal{EL}$++, as discussed in Sect. 5.3. $\mathcal{EL}$++ ontologies can be handled efficiently by the CEL reasoner [Baader et al., 2006].

Schema Graph The otological knowledge is represented as a graph that the query expansion algorithm can traverse to select terms. The graph built on the axioms in ontologies, named *schema graph*, contains the intensional knowledge of the domain, each vertex corresponds to a concept and each edge represents the relationship (roles) between concepts. For instance, an axiom of the form $A \sqsubseteq \exists R.B$ is represented as a subgraph where both A and B are vertices related by the ISA

relationship. Note that this *IS-A* relationship is only accessible from A to B, not vice versa. The inverse relationship *HAS-A* can be inserted from B to A to account for the other direction.

Taxonomic Candidates Centrality, i.e., the distance to the original concepts, is a crucial factor for determining candidate terms from ontologies, descended from the theory of *spreading activation* [Collins and Loftus, 1975]. The spreading activation algorithm starts from one or more concept nodes in a taxonomy, and activates all the nodes connected to each of them. When the spreading stops, the concepts on all the activation paths are considered candidate terms.

User studies in [Jones et al., 1995] showed that users seldome chose terms with a conceptual distance greater than 4. This fact serves as a reasonable stop condition for this query expansion algorithm, i.e., the selection of candiate terms can be limited within some *conceptual distances* when traversing the edges radiated from the central concepts [Rada and Bicknell, 1989; Croft et al., 1989; Biswas et al., 1986; Voorhees, 1994; Wollersheim and Rahayu, 2005b; Järvelin et al., 2001; Farfan et al., 2009; Dey et al., 2005].

Intuitively, the spreading activation algorithm traverse all the *IS-A* and *HAS-A* relations to move upwards and downwards hierarchically for superconcepts and subconcepts resp., and moves sideways seeking neighboring terms, e.g., synonyms and related terms, within the schema graph. Specifically, all the terms C_{T_i} that are superconcepts or subconcepts of C up to certain level i are considered candidate terms for C. Synonyms to C, i.e., terms on the same level as C or synset elements for WordNet concepts [Moldovan and Mihalcea, 2000], can also be added to the set of candidate terms. This traversal algorithm is efficient, but it has the disadvantage of losing the precise definitional information that makes the concept distinguishable, thus this traversal algorithm may introduce noisy concepts to the candidate set. Considering the following simple example, $Chair \sqsubseteq Professor \sqcap (\exists headOf.Department)$, where a chair can only be distinguished from other professors by observing her administrative role specified by the existential quantification. The traversal based on the above algorithm will expand the query keyword $Chair$ with terms that mention $Professor$ and $Department$. This may give rise to potential false positives that mention $Professor$ and $Department$ in a relation other than $headOf$, e.g., $worksIn$. To our knowledge, only a few traversing algorithms take existential quantification into consideration, e.g., [Farfan et al., 2009]. Furthermore, observe that universal quantification ($\forall$) is not exploited in existing works, primarily because it is beyond the syntax of typical DLs employed to model most medical ontologies, e.g., $\mathcal{EL}++$.

An adaptation of the above algorithm considers the quantified constraints. In the previous example, the algorithm selects the role $headOf$ as a candidate term. The rational behind such a strategy is similar to adding *context* words [Bai et al., 2007]. It is easy to reveal that role names are equally important as concepts. In this example, the concepts $Professor$ and $Department$ can ideally be related by a third candidate term, the role name $HeadOf$ or simply $Head$, to reflect the distinction between a professor and the chair. Named roles (relations) can be accessed and evaluated in many ways, according to the nature and characteristics of the relations, as discussed in Sect. 5.3.

In addition to explicit traversals on the schema graphs, negations can also be exploited, which was used by human experts as well [Bates, 1979]. For instance, the term $Speedy$ can be expanded as the negation to the term $Slow$. In a Wordnet/thesaurus this type of expansion, as a form of antonymy, is not uncommon, but such relationships in a DL represented ontology requires negations explicitely, e.g., $Speedy \sqsubseteq \neg Slow$. In fact, as empirically shown in [Wang et al., 1985], antonymy *degrades* the retrieval effectiveness, while the taxonomical terms tend to enhance the results to some extent.

Auxiliary Ontological Candidates The previous discussion about taxonomy traversal shows that named concepts and roles in the taxonomy can qualify as candidate terms. In addition to taxonomical concepts, other entities in an ontology may also be exploited, depending on the logical constructs allowed in the syntax of the underlying DLs of ontologies.

Firstly, the Abox instances constitute an *instance graph*, in addition to the schema graph built on TBox axioms. Individual names can be leveraged to expand the queries using the same strategy as for soliciting concepts, e.g, Dolby et al. [2009] instantiated query variables with individuals in different patterns depending on the query concepts, and Liu et al. [2009] instantiated all the concepts and store all the instances and relationship information in databases. A concrete example follows. When the concept $Canadian\_Prime\_Minister$ occurs in the query, the individual name $Stephen\_Harper$ may qualify as a candidate term because the individual instantiates the concept $Canadian\_Prime\_Minister$ in the ontology.

In spite of the possibility of using individual names, some technical issues pop up. First of all, the size of instance graph, compared to the schema graph, can be enormously large, which requires more efficient algorithms for traversing the graph. The exact number of instances for a particular concept may not be known due to the lack of appropriate indexing, e.g., in the setting of the Web, thus, only under very rare situations can a concept be instantiated to expand the query. If a concept has a large quantity of instances, it may be better off to leave the concept uninstantiated due to the sheer size. For example, instantiation of the query concept *Student* that enumerates all the students should be avoided. Another issue is brought up by the concretization of relations. Relations in schema graph express association between entity classes (concepts), while they denote associations between specific entities (individuals) in an instance graph. To allow the use of the concretized relations, the cardinality of relations, i.e., the number of participating members in both entity classes (for binary roles) should be considered. For example, $marriedTo$ is a one-to-one relation, $hasFather$ is a many-to-one relation, and $hasAuthor$ is a many-to-many relation. Such considerations appear more important in query answering systems, e.g., [Dolby et al., 2009; Liu et al., 2009; Pound et al., 2010], that heavily take advantage of individuals.

Concrete domain entities may also be eligible for candidate terms. Specifically, feature-value pairs that denote pivotal attributes of objects may be incorporated in query expansion. Semi-structured data often involves a significant number of features, which play a key role in a successful semantics-based query expansion. For instance, a publication record that has a list of authors can derive meaningful coauthorship information for a sample query asking for *collaborators of a particular researcher*. Features and values often suggest a precise contextual information for refining query expansion.

Other Algorithms for Term Selection Earlier research that uses thesaurus like WordNet have specific algorithms for obtaining candidate terms. For example, Navigli and Velardi [2003] constructed a semantic network for every keyword $\mathcal{K}_i$ in the query and every synset of $\mathcal{K}_i$ (i.e., every sense of $\mathcal{K}_i$ in WordNet). The creation of such semantic networks depends on the chosen relations, either drawn directly from WordNet or other sources. Note that the choice of relations is critic as some relations may introduce noises rather than useful candidates. For instance, gloss relations, as reported in Navigli and Velardi [2003], sometimes digress the subject from the query intent. Navigli and Velardi [2003] consequently provided two strategies to specify which keywords can be expanded. For example, one strategy allows only monosemous keywords to be expanded, ensuring a high precision, while the other allows keywords that closely relate to other keywords, in terms of a threshold number of common nodes in synsets, to be expanded.

Expansion on spatial queries involving geographical locations can exploit footprints [Fu et al., 2005] to select terms. That is, geographical locations in a query are translated into map coordinate footprint, only terms that are in the proximity of the footprints will be considered candidates, which avoids the introduction of possibly irrelevant query terms by conventional query expansion.

Non-ontological Entity Selection Existing query expansion strategies[18] at large only consider the ontologies as the sources of candidate terms [Díaz-Galiano et al., 2009; Moldovan and Mihalcea, 2000]. When the ontologies are drawn from the corpus, query expansion is in fact using part of the document collection information as ontologies capture most salient terms in the documents. Some salient terms residing in the documents may be missing in the ontologies, thus query expansion may benefit from soliciting candidate terms from multiple sources, including the document collection [Qiu and Frei, 1993; Grootjen and van der Weide, 2006] and the feedback data (discussed in Sect. 8). Indeed, Srinivasan [1996c] described three solicitation strategies, i.e., adding document terms to the original query, adding ontological concepts to the original query, and a combination of the previous two strategies. In addition, it is possible to cluster the terms in the collection, consequently a keyword only finds its candidates from the same cluster, as seen in [Wang et al., 2009; Grootjen and van der Weide, 2006]. Every cluster can be viewed a concept, uniquely identified by the set of terms in it [Grootjen and van der Weide, 2006]. Similarly, Google Sets[19] identifies groups of related items on the web based on the input of several terms. The advantage of Google Sets lies in its utilization of the largest corpus (i.e., the Web) to derive related terms, so the accuracy is high on average. The weakness also springs from the web: the generated terms may be related in a more broad sense, bringing in noises for term suggestion in a specific domain.

As observed from the information specialists, the tactic *provenance* in Table 1 can be leveraged in particular. That is, it may be more effective to use candidate terms generated from an initially retrieved set of documents instead of the whole corpus [Chang et al., 2006; Grootjen and van der Weide, 2006][20]. Retrieval feedback [Srinivasan, 1996c] uses a feedback-driven strategy to solicit candidate terms. This strategy explores the top few documents returned by the original query, then it treats *all* terms in these documents as candidate terms. As terms are generated from the initially retrieved documents by the original query, the terms may not provide additional information that the original query lacks. A solution presented in [Grootjen and van der Weide, 2006] to the above weakness is to generate a formal concept derived from relevant documents in the whole collection. Intuitively, this solution integrates global analysis into local analysis to minimize the side effect due to local relevance feedback.

Quantity of Candidate Terms User participation in [Jones et al., 1995] indicates that it is better to provide plenty of candidate terms for real users to choose from. A potential problem of selecting as many terms as possible is that the time required for query evaluation increases as well. Furthermore, some selected terms may be harmful or useless [Cao et al., 2008], especially for automatic terms selection without human intervention.

Most experiments evaluated the effects of the quantity of terms added to a query under specific situations. Empirical studies on relevance feedback suggest that "doubling the number of new terms

[18] Ontology-based query answering systems [Dolby et al., 2009; Pound et al., 2010] only need to use ontological entities.

[19] http://labs.google.com/sets

[20] However, a snag in [Chang et al., 2006] is that these terms were used for query reformulation, not for query expansion.

will add a constant to the recall-precision measure" and the effectiveness ceases increasing after a certain number of terms have been added [Buckley et al., 1994a]. Mandala et al. [2000] empirically showed that retrieval improvement increases till more than 20 terms were added, then it levels off. More added terms, say up to 50, tend to deteriorate the improvement for the collection in their experiment setting. For concept-based query expansion, Qiu and Frei [1993] pointed out that adding as much as 100 top-ranked terms seems "to be the safe way to go", yet the number of candidate terms is reduced to 20 when relevance feedback is used because of the smaller number of documents returned by feedback. Jing and Croft [1994] concluded that the number of terms to be added to a query depends on many factors involving the nature of the query and the candidate terms, the size of the collection and so on. How to determine this number for any given query is not clear. According to [Cao et al., 2008], the retrieval effectiveness with a reasonably small number of candidate term decreased slightly, but still much higher than the baseline approaches. Consequently, it is better to parametrize the quantity of selected terms to allow for efficient evaluation.

6.2 Ranking Candidate Terms

It is assumed that the "best" candidate terms used to expand the query will effectively improve the retrieval results. The selection of the most promising one(s) from the set of candidate terms is however not very obvious. Some ranking criteria can be devised to aid the seection. Ranking functions may be subjective and can only be empirically verified, so far there is no universal ranking functions for evaluating candidate terms in different domains and datasets. Informally, candidate terms that are more *similar* to the query concepts are likely to capture the conceptual idea behind the concept. For traditional keyword query expansion, the similarity between a candidate term and one or more keywords is computed, while for concept based query expansion, the similarity is calculated between a candidate term and the query concept [Qiu and Frei, 1993]. We think the discrepancy in similarity computation is not very significant for the above two, and most measures can be applied to both. In what follows, we elucidate some commonly applied measures for computing the similarity between a candidate term and the original concept, which may compute the structural similarity (syntactic measures), the semantic similarity (semantic measures), and other contextual similarity. Regarding this paper, semantic similarity is more important for deciding the similarity distances between terms.

Syntactic Measures A prevalent structural comparison between candidate terms and the original concept is to check the syntactic difference, e.g., [Pu and Yu, 2008]. In simple cases, terms are viewed isolated. For a term τ and a query concept c, the *edit distance*, i.e., the minimum number of operations needed to transform τ to c, can be computed using existing algorithms, e.g., [Manning et al., 2008, Chap. 3]. In addition, operations, e.g., insertion, deletion, replacing, etc., can be weighted. More advanced algorithms make use of other techniques, for example, the k-gram index method computes the overlap between the set of k-grams in τ and in c to reflect the distance between τ and c. If terms are considered context-sensitive, semantic measures are favored.

Traditional IR measures, e.g., *df* and *tf-idf* for rank-ordering documents, are also applicable for computing term similarity. For instance, Farfan et al. [2009] viewed XML elements as documents to apply *df* for computing the association degree of a node to a keyword. Qiu and Frei [1993] also used these statistical measures with the roles of terms and documents interchanged. Wang et al. [2009] adopted this idea by viewing ads as documents and bid phrases as terms. In addition to co-occrrence

based measures, other syntactic measures can be used, e.g., [Croft, 1986], where probability values are attached to inference rule to rank the terms recognized by the rules.

Semantic Measures Semantic measures take advantage of the semantics existing in a knowledge base, including commonly seen taxonomy-based measures and other advanced techniques. There exist a corpus of literature on semantic similarity, e.g., [Collins and Loftus, 1975; Resnik, 1995; Rada et al., 1989; Maguitman et al., 2005; Schenkel et al., 2005; Theobald, 2003; Hersh et al., 2000], for interested readers. Observe that semantic similarity can be determined by collecting evidence to exceed some positive or negative criterion [Collins and Loftus, 1975]. In most existing approaches, the positive criterion is validated. Indeed the negative criterion typically concludes that a concept is not similar to another.

Taxonomic Measures A classic metric for measuring semantic similarity is conceptual distance [Rada et al., 1989], originated from spreading activation. Conceptual distance (a.k.a. centrality) suggests how much is a destination concept removed from the *origin* concept(s) on the schema graph, and the shortest path length is used to measure conceptual distance, as seen in [Tudhope et al., 2006; Voorhees, 1994; Wollersheim and Rahayu, 2005b; Järvelin et al., 2001; Farfan et al., 2009; Schenkel et al., 2005; Lee et al., 2008]. Rada and Bicknell [1989] heavily relied on broader terms as real users have the tendency to use them. Interactive studies in [Jones et al., 1995] showed that users prefer narrower and related terms over broader terms for the specific test suite, contrary to the previous findings. Jones et al. [1995] analyzed the user choices of terms in different conceptual distances, and it seemed that candidate terms of shorter distances tend to contribute more to successful query enhancement, which confirms the legitimate use of conceptual distance as a measure of term importance.

To begin with the adapted spreading activation algorithm, the ontological entities matched to the query terms are marked as origin concepts, known as the starting point of the algorithm. By convention, the weight assigned to every candidate term is in the range [0,1]. Origin concepts are assigned an initial weight of 1. To account for the conceptual distances, a *decay factor* is used to reflect the weight reduction of the destination concepts in traversal. The value of the decay factor may vary from application to application, dependent on the initial query and the schema graph.

Other than distance, the number of relations that relate other concepts to the target concept also affect the target concept's weight., i.e., the *fan-in* and *fan-out* of nodes in the schema graph, as shown in [Wollersheim and Rahayu, 2005a,b; Farfan et al., 2009]. The intuition is that terms with higher degrees, being more easily accessible than those with lower degrees, appear to be more distracting and less useful.

The *criteriality of relations* reflects the significance of different relations instead of treating all edges the same [Kim and Kim, 1990; Tudhope et al., 2006; Schenkel et al., 2005]. For example, *IS-A* is likely to be assigned higher weight than the relation *IS-NOT-A* that indicates negative similarity [Rada et al., 1989]. Though Jones et al. [1995] gave possible criteria for weight assignment, e.g., the number of connections to the relevant term, the authors also pointed out that there is no influential relationships for users to select terms, therefore there is *little* evidence to justify the different weights of relationships. Indeed, as reported in [Kim and Kim, 1990], assigning weights to relationships was subjective and difficult, and may cause inconsistency in a large-scale taxonomy.

Eq. 1 describes a possible ranking function for computing the weight $w(D_m)$ of some concept D_m based on the previous factors, where D_m is related to D_n by the relationship rel_n^i on path i. It is assumed that there are k paths that start from some origin concept D_0 leading to D_m in

Eq. 1. Additionally, $\mathcal{D}ecay$ and $\mathcal{W}_r(rel_n^i)$ are the decay factor and the criteriality of relationship respectively. $\lambda(k)$ is a function that measures the fan-in degree (k) of concepts, which can also take other factors into consideration, e.g., the length of each fan-in path. Observe that the weight for a candidate term can go beyond the range because of the additive property in weight computing. Consequently, the set of n candidate terms should have their weight normalized, e.g., Eq. 2, where $\mathcal{W}(\cdot)$ is overloaded to represent the weight of all elements in the vector.

$$\mathcal{W}(D_m) = \begin{cases} 1 & \text{if } m = 0, \\ \lambda(k) \cdot \sum_{i=1}^{k} \mathcal{W}(D_n) \times \mathcal{D}ecay \times \mathcal{W}_r(rel_n^i) & \text{otherwise;} \end{cases} \tag{1}$$

$$\mathcal{W}\hat{(D)} = \frac{\overrightarrow{\mathcal{W}(D)}}{\sqrt{\sum_{i=1}^{n}(\mathcal{W}(D_i))^2}} \tag{2}$$

When entities in the schema graph carry weight (e.g., the certainty or confidence of the entities) a prior, the final weight should reflect that weight as well [Rada and Bicknell, 1989; Kim and Kim, 1990]. Smoothness should be preserved under such circumstances (see Sect. 7.1), e.g., for any two entities with (dis)similar prior weight, they also have (dis)similar final weight. Regarding computing the differences between a positive concept A and a negation $\neg B$, Rada et al. [1989] considered the conceptual distance as that between A and the *set* of concepts that are farthest from B in the schema graph. Kim and Kim [1990] extended the idea, and defined the negation context subgraph for $\neg B$, where the context for $\neg B$ can be retained and a substitution set can be found for distance computation. Details are available in [Kim and Kim, 1990].

Supplementary Semantic Measures As argued by Tversky [1977], object similarity, different from dimensional or metric measures, can be computed by a feature-matching process. Functional roles (denoted as $\mathcal{F}$ in DLs) can model the well-known *attributes* or *features* of entities, typically concepts. Specifically, a linear combination of the common and distinctive features describing concepts or terms offers similarity judgement.

ABox information, where applicable, can be utilized to compute similarity. For instance, if a term corresponds with a concept that has instances in the knowledge base, the number of its instances that also belong to the original concept partly suggest how related the term is to the original concept [Pound et al., 2010]. Analogously, the facets (subjects) that a concept share in common with others are another indicator of semantic similarity. Note that both the instances of a concept and the facets that a concept belongs to can be thought of as features of the concept. More advanced techniques, e.g., information-theoretic measures of similarity [Resnik, 1995; Maguitman et al., 2005], are also applicable but beyond the scope of this paper.

Remarks on Similarity Measures Grootjen and van der Weide [2006] proposed that practically human intervention is necessary to locate an optimal candidate concept for expansion. In their approach the users have to navigate through the local thesaurus to determine the best concept, which minimizes the query ambiguity but often has a negative effect on user experience.

In some cases, the ranking of candidate terms depends on the algorithm that selects them. For instance in Navigli and Velardi [2003], senses, given the semantic networks, are intersected pairwise, and the common nodes shared by both semantic networks in each intersection are totaled. Every sense configuration (i.e., one sense per keyword for all keywords in a configuration) can now be scored in terms of the number of common nodes. Candidate terms are drawn in the top-scored

configuration by five selection methods, for example, a keyword may be expanded by its synset in the "best" configuration.

There also exist some general approaches for ranking. For instance, the use of search engines may determine the rank order between terms. Sample queries consisting of the original concept and a candidate term can be run in a search engine. The number of the returned webpage hits suggests the association degree between the concept and the candidate term [Theobald, 2003]. Such approach is independent of ontologies and domains, yet the domain knowledge is underspecified.

7 Post-processing

Given the selected candidate terms, we are generally concerned with the *overall* effects of the set of terms, while overlooking the fact whether every individual term in the set is indeed useful for improving retrieval. It might be the case that the global retrieval effectiveness can be enhanced while a significant portion of the expansion terms is useless [Cao et al., 2008], given a particular query. The robustness of query expansion may be degraded too, i.e., the number of queries whose effectiveness are hurt due to the expansion may increase. There is consequently necessity to optimize the candidate terms in order to minimize the negative impact by expanded queries. Existing works [Metzler and Croft, 2007; Collins-Thompson, 2009] already attended to the robustness of query expansion approaches, as discussed in Sect. 7.1. Once optimized, the set of refined candidate terms is added to the original query, as shown in Sect. 7.2.

7.1 Optimization

The variables that affect the retrieval quality roughly fall into two dimensions. One is related to single user profiles, e.g., the user goal, user search experience, the domain knowledge of a user and so on. Individual context including the aforementioned factors is very challenging to extract from implicit user feedback, e.g., user logs [Sihvonen and Vakkari, 2004; Bai et al., 2007; Duggan and Payne, 2008]. Another observation is that the user profile may bias the queries unrelated to the profile [Bai et al., 2007]. The other dimension focuses on the queries. The query-centric variables are generally objective and easy to quantify. Table 2 describes the variables to be optimized for terms obtained by ontology-based QE approaches. The two sets of variables in *source coverage* and *facet coverage* are described in details, while the other variables are self-evident.

Source coverage typically reflects how a QE algorithm respond to all the available options. Naturally, an algorithm should perform a global analysis (i.e., a collection of *multiple evidences*) of all the options instead of choosing most terms from a particular source prematurely. Conversely, the portion of candidate terms selected from a source indicates the influence of that source, allowing for fine-tuning the weights of sources.

Facet coverage and *facet focus* are mutually exclusive. The former requires terms be balanced on all facets, which aims to meet users' information need by diversifying the term aspects. Contrarily, the latter is biased towards certain facets. Some facets may convey much more of the query intent than the others; some facets may be more obscure to the search engine than the others such that the facets need more terms for elucidation. This was substantiated in [Sihvonen and Vakkari, 2004] that the number of terms varied between facets for QE.

Source coverage	`The ratio of the utilized sources to all the available sources; the ratio of the selected terms to all the available terms in one source.`
Query intent	`The selected terms should relate to the query concept, not individual query terms, to preserve query intent.`
Facet coverage	`The selected terms spread over all the facets in query intent; the selected terms focus more on articulating certain facets in the query.`
Quantity	`The number of selected terms to expand the query.`

Table 2. Optimization Variables

Optimization Frameworks Every smoothing or optimization technique applied to the selected terms has to balance between two aspects, one being the closeness of the optimized term weights to the original ones and the other being the neighboring smoothness of term weights [Mei et al., 2008]. A reliable and effective framework for optimizing candidate terms was presented in [Collins-Thompson, 2009], where the variables in Table 2 can be defined as constraints for optimization. In fact, all but the first set of variables in Table 2 were already used in [Collins-Thompson, 2009]. A complete description of the optimization framework is beyond the scope of this paper.

7.2 Query Formulation

Expert users employ advanced techniques to formulate queries, when they have acquired a number of additional terms for expansion. A successful expansion consequently also depends on the way the candidate terms are arranged to expanded the original queries. When formulating a query, human experts, on the one hand, want terms to be as *specific* as possible to eliminate a large part of a corpus. On the other hand, experts *exhaust* the terms for expressing the query. Additionally, the number of these terms is kept to a minimum to *reduce* noises. A similar strategy can be adopted by an automatic QE system to formulate queries. Consider a boolean keyword search engine that supports conjunctive (`AND`), disjunction(`OR`) and negation (`NOT`) operators. In general, terms added in conjunction and negation with the original keywords tend to enhance precision [Greenberg, 2001b; Lee et al., 2008], while terms added in disjunction lead to a higher recall, as in [Navigli and Velardi, 2003; Järvelin et al., 2001; Chang et al., 2006; Greenberg, 2001a].

Because query expansion is normally implemented as a recall-enhancing technique, precision may decrease due to the possible noises introduced by the expanded terms. For instance, Greenberg [2001b] showed in the experiments that synonyms and subconcepts augmented recall with a slight loss in precision. Superconcepts and related terms (e.g., via the relationship *related_to*) are also useful when high recall is desired, but such terms have a significant detrimental effect on retrieval precision.

Structured queries using conjunctions and disjunctions in a predetermined way for query formulation is possible. For instance, [Sihvonen and Vakkari, 2004] provided boxes for facets such that terms within a box form disjunctions and terms between boxes form conjunctions. Nevertheless,

the choice of the final form of an expanded query depends on the user's specific information need. Undoubtedly, an interactive environment is of great assistance to query formulation.

8 Discussion and Conclusions

Complementing the Search Strategy This paper proposes a synthesized plan for ontology-based query expansion, that is, candidate terms are selected and ranked in both the syntactic and the semantic dimensions. In Sect. 1.1, we already mentioned that ontology-based QE systems can integrate other search tactics to obtain optimal retrieval results, as did [Srinivasan, 1996c; Aronson and Rindflesch, 1997]. Indeed, Bates [1979] emphasized that in many cases "user feedback during the search adds another dimension of complexity to the search." This idea of exploiting user feedback has been practiced by various approaches. Relevance feedback, as discussed in Sect. 1.1, already showed great potential for enhanced retrieval. Moreover, White et al. [2005] established the occasions when the utility of implicit relevance feedback is promising. A common approach is to use both judgements in the search strategy, whereas the details are beyond the discussion of this paper.

Document Ranking The classic vector space model, see [Manning et al., 2008, Chap. 6], is generally employed for ranking retrieved documents. [Voorhees, 1994; Castells et al., 2007] computes the weighted sum of a document vector of terms and an extended query vector of candidate terms. Systems that integrate knowledge bases in document retrieval, e.g., [Biswas et al., 1987], rank documents based on multple evidences that combine both statistical and semantic/knowledge-based similarity measures. For example, Biswas et al. [1987] defined similarity measures based on the Dempster-Shafer theory of evidence combination, which reflect the process of belief revision and updating; Croft et al. [1989] combined term-based, nearest-neighbor and citation evidence to assess the overall document relevance.

Relevance ranking involves subjective judgment. Given the same query, a relevant document to one user may be irrelevant to another user. Consequently, research that exploits the user behavior tend to complement document ranking as well.

Performance Evaluation Mandala et al. [2000] compared relevance feedback to ontology-aided query expansion, and proclaimed that the later outperforms pseudo relevance feedback remarkably but was slightly less effective than *ideal* relevance feedback. Srinivasan [1996c] reported significant improvements using different expansion strategies. It further showed that expansions that the strategies using MeSH ontology yielded better improvements than those disregarding ontologies. However, Hersh and Hickam [1995] mentioned that the benefit of using MeSH terms depends on the user: only users well-trained in manipulating such domain terms contribute to improved retrieval results.

The experiments in [Wollersheim and Rahayu, 2005b] showed that the expansion algorithm that used semantic content in the query achieved the best performance. Concept-based query expansion yields more desirable results, but the downside is that the performance for queries of a single term may be degraded. In addition, concept-based expansion algorithm does not work for word-based queries, where most terms in a query are too semantically distant to form a concept. Aronson et al. [1994] showed that an altered query with only concepts (i.e., the keywords that don't correspond to any concept are removed) had a detrimental effect on the retrieval results, yet expanded queries retaining original keywords are more useful. The average precision improves over the plain text in [Aronson et al., 1994] is only 4%.

Although various experiments have been carried out to show how ontologies can be exploited to expand original queries and to enhance retrieval performance, the conclusions drawn from these empirical studies were inconsistent, and mostly rule of thumb. Typically, we have not yet witnessed any experimental system that employ ontologies for query enhancement to consistently improve retrieval performance. A recommendation is that a synthesized query expansion plan (Sect. 3) that absorbs previous experience (Sections 4 and 6) using a well-built domain ontology (Sect. 5) can yield better retrieval performance.

Bibliography

Alan R. Aronson. Effective mapping of biomedical text to the UMLS metathesaurus: the MetaMap program. In *Proceedings of AMIA, Annual Symposium*, pages 17–21, 2001.

Alan R. Aronson and Thomas C. Rindflesch. Query expansion using the UMLS Metathesaurus. In *Proceedings of the AMIA Annual Fall Symposium*, pages 485–489, 1997.

Alan R. Aronson, Rindflesch C. Thomas, and Browne C. Allen. Exploiting a large thesaurus for information retrieval. In *Proceedings of RIAO*, pages 197–216, 1994.

F. Baader, C. Lutz, and B. Suntisrivaraporn. CEL—a polynomial-time reasoner for life science ontologies. In U. Furbach and N. Shankar, editors, *Proceedings of the 3rd International Joint Conference on Automated Reasoning (IJCAR'06)*, volume 4130 of *Lecture Notes in Artificial Intelligence*, pages 287–291. Springer-Verlag, 2006.

Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003. ISBN 0521781760.

Jing Bai, Jian-Yun Nie, Guihong Cao, and Hugues Bouchard. Using query contexts in information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 15–22, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-597-7. doi: http://doi.acm.org/10.1145/1277741.1277747.

Holger Bast, Alexandru Chitea, Fabian Suchanek, and Ingmar Weber. Ester: efficient search on text, entities, and relations. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 671–678, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-597-7. doi: http://doi.acm.org/10.1145/1277741.1277856.

M. J. Bates. Information search tactics. *Journal of the American Society for Information Science*, 30(4):205–214, 1979.

J. Bhogal, A. Macfarlane, and P. Smith. A review of ontology based query expansion. *Information Processing and Management*, 43(4):866–886, 2007. ISSN 0306-4573. doi: http://dx.doi.org/10.1016/j.ipm.2006.09.003.

Chris Biemann. Ontology learning from text: a survey of methods. *LDV-Forum*, 20(2):75–93, 2005.

G Biswas, J C Bezdek, and R L Oakman. A knowledge-based approach to online document retrieval system design. In *Proceedings of the ACM SIGART international symposium on Methodologies for intelligent systems*, pages 112–120, New York, NY, USA, 1986. ACM. ISBN 0-89791-206-3. doi: http://doi.acm.org/10.1145/12808.12821.

Gautam Biswas, James C. Bezdek, Viswanath Subramanian, and Marisol Marques. Knowledge-assisted document retrieval: II. the retrieval process. *Journal of the American Society for Information Science*, 38(2):97C110, 1987.

O. Bodenreider. Biomedical ontologies in action: role in knowledge management, data integration and decision support. *Yearbook of medical informatics*, pages 67–79, 2008. ISSN 0943-4747. URL `http://view.ncbi.nlm.nih.gov/pubmed/18660879`.

Falk Brauer, Michael Huber, Gregor Hackenbroich, Ulf Leser, Felix Naumann, and Wojciech M. Barczynski. Graph-based concept identification and disambiguation for enterprise search. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 171–180, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-799-8. doi: http://doi.acm.org/10.1145/1772690.1772709.

Christopher Brewster and Yorick Wilks. Ontologies, taxonomies, thesauri learning from texts. In Marilyn Deegan, editor, *Proceedings of the Workshop on the Use of Computational Linguistics in the Extraction of Keyword Information from Digital Library Content*. Centre for Computing in the Humanities, Kings College London, 2004. 5-6 February.

Chris Buckley, Gerard Salton, and James Allan. The effect of adding relevance information in a relevance feedback environment. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA, 1994a. Springer-Verlag New York, Inc. ISBN 038719889X.

Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. Automatic query expansion using smart: Trec 3. In *TREC*, 1994b.

Michael J. Cafarella, Christopher Ré, Dan Suciu, Oren Etzioni, and Michele Banko. Structured querying of web text: A technical challenge. In *CIDR*, 2007. URL `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.100.3465`.

Silvia Calegari and Gabriella Pasi. Personalized ontology-based query expansion. *Web Intelligence and Intelligent Agent Technology, IEEE/WIC/ACM International Conference on*, 3:256–259, 2008.

Guihong Cao, Jian-Yun Nie, Jianfeng Gao, and Stephen Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 243–250, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-164-4. doi: http://doi.acm.org/10.1145/1390334.1390377.

Pablo Castells, Miriam Fernandez, and David Vallet. An adaptation of the vector-space model for ontology-based information retrieval. *IEEE Trans. on Knowl. and Data Eng.*, 19(2):261–272, 2007. ISSN 1041-4347. doi: http://dx.doi.org/10.1109/TKDE.2007.22.

Soumen Chakrabarti, Byron Dom, Rakesh Agrawal, and Prabhakar Raghavan. Scalable feature selection, classification and signature generation for organizing large text databases into hierarchical topic taxonomies. *The VLDB Journal*, 7(3):163–178, 1998. ISSN 1066-8888. doi: http://dx.doi.org/10.1007/s007780050061.

Youjin Chang, Iadh Ounis, and Minkoo Kim. Query reformulation using automatically generated query concepts from a document space. *Inf. Process. Manage.*, 42(2):453–468, 2006. ISSN 0306-4573. doi: http://dx.doi.org/10.1016/j.ipm.2005.03.025.

Tao Cheng, Xifeng Yan, and Kevin Chen-Chuan Chang. Entityrank: searching entities directly and holistically. In *VLDB '07: Proceedings of the 33rd international conference on Very large data bases*, pages 387–398. VLDB Endowment, 2007. ISBN 978-1-59593-649-3.

Allan M. Collins and Elizabeth F. Loftus. A spreading-activation theory of semantic processing. *Psychological Review*, 82(6):407 – 428, 1975. ISSN 0033-295X.

Kevyn Collins-Thompson. Reducing the risk of query expansion via robust constrained optimization. In *CIKM*, pages 837–846, 2009.

W. B. Croft. User-specified domain knowledge for document retrieval. In *SIGIR '86: Proceedings of the 9th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 201–206, New York, NY, USA, 1986. ACM. ISBN 0-89791-187-3. doi: http://doi.acm.org/10.1145/253168.253211.

W. B. Croft, T. J. Lucia, J. Cringean, and P. Willett. Retrieving documents by plausible inference: an experimental study. *Inf. Process. Manage.*, 25(6):599–614, 1989. ISSN 0306-4573. doi: http://dx.doi.org/10.1016/0306-4573(89)90095-2.

C. J. Crouch. An approach to the automatic construction of global thesauri. *Inf. Process. Manage.*, 26(5):629–640, 1990. ISSN 0306-4573. doi: http://dx.doi.org/10.1016/0306-4573(90)90106-C.

Carolyn J. Crouch and Bokyung Yang. Experiments in automatic statistical thesaurus construction. In *SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 77–88, New York, NY, USA, 1992. ACM. ISBN 0-89791-523-2. doi: http://doi.acm.org/10.1145/133160.133180.

Hang Cui, Ji-Rong Wen, Jian-Yun Nie, and Wei-Ying Ma. Probabilistic query expansion using query logs. In *WWW '02: Proceedings of the 11th international conference on World Wide Web*, pages 325–332. ACM Press, 2002.

Wisam Dakka and Panagiotis G. Ipeirotis. Automatic extraction of useful facet hierarchies from text databases. In *ICDE 08: Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, pages 466–475, Washington, DC, USA, 2008. IEEE Computer Society. ISBN 978-1-4244-1836-7. doi: http://dx.doi.org/10.1109/ICDE.2008.4497455.

Nilesh Dalvi, Ravi Kumar, Bo Pang, Raghu Ramakrishnan, Andrew Tomkins, Philip Bohannon, Sathiya Keerthi, and Srujana Merugu. A web of concepts. In *Proceedings of the twenty-eighth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, PODS '09, pages 1–12, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-553-6. doi: http://doi.acm.org/10.1145/1559795.1559797.

Pedro DeRose, Warren Shen, Fei Chen, AnHai Doan, and Raghu Ramakrishnan. Building structured web community portals: a top-down, compositional, and incremental approach. In *Proceedings of the 33rd international conference on Very large data bases*, VLDB '07, pages 399–410. VLDB Endowment, 2007. ISBN 978-1-59593-649-3.

Lipika Dey, Shailendra Singh, Romi Rai, and Saurabh Gupta. Ontology aided query expansion for retrieving relevant texts. In *Advances in Web Intelligence Third International Atlantic Web Intelligence Conference*, pages 126–132, 2005.

M. C. Díaz-Galiano, M.T Martín-Valdivia, and L. A. Ureña López. Query expansion with a medical ontology to improve a multimodal information retrieval system. *Computers in Biology and Medicine*, 39(4):396–403, 2009. ISSN 0010-4825. doi: http://dx.doi.org/10.1016/j.compbiomed.2009.01.012.

Stephen Dill, Nadav Eiron, David Gibson, Daniel Gruhl, R. Guha, Anant Jhingran, Tapas Kanungo, Sridhar Rajagopalan, Andrew Tomkins, John A. Tomlin, and Jason Y. Zien. Semtag and seeker: bootstrapping the semantic web via automated semantic annotation. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 178–186, New York, NY, USA, 2003. ACM. ISBN 1-58113-680-3. doi: http://doi.acm.org/10.1145/775152.775178.

Julian Dolby, Achille Fokoue, Aditya Kalyanpur, Edith Schonberg, and Kavitha Srinivas. Efficient reasoning on large SHIN aboxes in relational databases. In *Proceedings of the 5th International Workshop on Scalable Semantic Web knowledge Base Systems (SSWS2009)*, pages 110–125, 2009.

Geoffrey B. Duggan and Stephen J. Payne. Knowledge in the head and on the web: using topic expertise to aid search. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, pages 39–48, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-011-1. doi: http://doi.acm.org/10.1145/1357054.1357062.

Susan Dumais, Edward Cutrell, JJ Cadiz, Gavin Jancke, Raman Sarin, and Daniel C. Robbins. Stuff I've seen: a system for personal information retrieval and re-use. In *SIGIR'03: Proceedings of the 26th ACM SIGIR conference on Research and development in informaion retrieval*, pages 72–79, New York, NY, USA, 2003. ACM. ISBN 1-58113-646-3. doi: http://doi.acm.org/10.1145/860435.860451.

Ofer Egozi, Evgeniy Gabrilovich, and Shaul Markovitch. Concept-based feature generation and selection for information retrieval. In *Proceedings of the 23rd national conference on Artificial intelligence - Volume 2*, pages 1132–1137. AAAI Press, 2008. ISBN 978-1-57735-368-3.

Fernando Farfan, Vagelis Hristidis, Anand Ranganathan, and MD Michael Weiner. XOntoRank: Ontology-aware search of electronic medical records. In *Proceedings of the 25th International Conference on Data Engineering*, pages 820–831, Shanghai, China, 2009. doi: http://doi.ieeecomputersociety.org/10.1109/ICDE.2009.73.

Gaihua Fu, Christopher B. Jones, and Alia I. Abdelmoty. Ontology-based spatial query expansion in information retrieval. In Robert Meersman and Zahir Tari, editors, *ODBASE: OTM Confederated International Conferences*, volume 3761 / 2005. Springer Berlin / Heidelberg, November 2005.

Rebecca Green. Relationships in the organization of knowledge: an overview. In Carol A. Bean and Rebecca Green, editors, *Relationships in the organization of knowledge*, chapter 1, pages 3–18. Kluwer Academic Publishers, New York, 2001.

Jane Greenberg. Automatic query expansion via lexical-semantic relationships. *J. Am. Soc. Inf. Sci. Technol.*, 52(5):402–415, 2001a. ISSN 1532-2882. doi: http://dx.doi.org/10.1002/1532-2890(2001)9999:9999¡::AID-ASI1089¿3.3.CO;2-B.

Jane Greenberg. Optimal query expansion (QE) processing methods with semantically encoded structured thesauri terminology. *Journal of the American Society for Information Science and Technology (JASIST)*, 52(6):487–498, 2001b.

F. A. Grootjen and Th. P. van der Weide. Conceptual query expansion. *Data Knowl. Eng.*, 56(2): 174–193, 2006. ISSN 0169-023X. doi: http://dx.doi.org/10.1016/j.datak.2005.03.006.

R. Guha and R. McCool. TAP: A semantic web test-bed. *Web Semantics: Science, Services and Agents on the World Wide Web*, 1(1):81 – 87, 2003. ISSN 1570-8268. doi: DOI: 10.1016/j.websem.2003.07.004.

W R Hersh, D H Hickam, and T J Leone. Words, concepts, or both: optimal indexing units for automated information retrieval. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*, pages 644–648, 1992.

William Hersh, Chris Buckley, T. J. Leone, and David Hickam. Ohsumed: an interactive retrieval evaluation and new large test collection for research. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 192–201, New York, NY, USA, 1994. Springer-Verlag New York, Inc. ISBN 0-387-19889-X.

William Hersh, Susan Price, and Larry Donohoe. Assessing thesaurus-based query expansion using the UMLS metathesaurus. In *In Proc. of the 2000 American Medical Informatics Association (AMIA) Symposium*, pages 344–348, 2000.

William R. Hersh and David Hickam. Information retrieval in medicine: The SAPHIRE experience. *Journal of the American Society for Information Science*, 46:743–747, 1995.

Hanh Huu Hoang and A Min Tjoa. The state of the art of ontology-based query systems: A comparison of existing approaches. In *Proceedings of ICOCI06 - The IEEE International Conference on Computing and Informatics*, 2006.

Ingrid Hsieh-Yee. Effects of search experience and subject knowledge on the search tactics of novice and experienced searchers. *JASIS*, 44(3):161–174, 1993.

Kalervo Järvelin, Jaana Kekäläinen, and Timo Niemi. Expansiontool: Concept-based query expansion and construction. *Information Retrieval*, 4(3-4):231–255, 2001. ISSN 1386-4564. doi: http://dx.doi.org/10.1023/A:1011998222190.

Yufeng Jing and Bruce Croft. An association thesaurus for information retrieval. In *RIAO*, pages 146–161, 1994.

Susan Jones, Mike Gatford, Steve Robertson, Micheline Hancock-Beaulieu, Judith Secker, and Steve Walker. Interactive thesaurus navigation: intelligence rules ok? *J. Am. Soc. Inf. Sci.*, 46(1):53–59, 1995. ISSN 0002-8231.

Gjergji Kasneci, Fabian M. Suchanek, Georgiana Ifrim, Shady Elbassuoni, Maya Ramanath, and Gerhard Weikum. Naga: harvesting, searching and ranking knowledge. In *SIGMOD '08: Proceedings of the 2008 ACM SIGMOD*, pages 1285–1288, New York, NY, USA, 2008. ACM.

Young Whan Kim and Jin H. Kim. A model of knowledge based information retrieval with hierarchical concept. *Journal of Documentation*, 46(2):113–136, 1990. ISSN 0022-0418. doi: http://dx.doi.org/10.1108/eb026857.

Atanas Kiryakov, Borislav Popov, Ivan Terziev, Dimitar Manov, and Damyan Ognyanoff. Semantic annotation, indexing, and retrieval. *Journal of Web Semantics*, 2(1):49–79, 2004. ISSN 1570-8268. doi: http://dx.doi.org/10.1016/j.websem.2004.07.005.

Ming-Che Lee, Kun Hua Tsai, and Tzone I. Wang. A practical ontology query expansion algorithm for semantic-aware learning objects retrieval. *Comput. Educ.*, 50(4):1240–1257, 2008. ISSN 0360-1315.

Shengping Liu, Yuan Ni, Jing Mei, Hanyu Li, Guotong Xie, Gang Hu, Haifeng Liu, Xueqiao Hou, and Yue Pan. ismart: Ontology-based semantic query of cda documents. In *AMIA Annu Symp Proc. 2009*, pages 375–379. American Medical Informatics Association, 2009.

Ana G. Maguitman, Filippo Menczer, Heather Roinestad, and Alessandro Vespignani. Algorithmic detection of semantic similarity. In *WWW '05: Proceedings of the 14th international conference on World Wide Web*, pages 107–116, New York, NY, USA, 2005. ACM. ISBN 1-59593-046-9. doi: http://doi.acm.org/10.1145/1060745.1060765.

Rila Mandala, Takenobu Tokunaga, and Hozumi Tanaka. Query expansion using heterogeneous thesauri. *Information Processing & Management*, 36(3):361 – 378, 2000. ISSN 0306-4573. doi: DOI: 10.1016/S0306-4573(99)00068-0.

Christopher D. Manning and Heinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, 1999.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 1 edition, July 2008. ISBN 0521865719.

Karen Markey. Twenty-five years of end-user searching, part 2: Future research directions. *J. Am. Soc. Inf. Sci. Technol.*, 58:1123–1130, June 2007. ISSN 1532-2882. doi: 10.1002/asi.v58:8.

Qiaozhu Mei, Duo Zhang, and ChengXiang Zhai. A general optimization framework for smoothing language models on graph structures. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 611–618, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-164-4. doi: http://doi.acm.org/10.1145/1390334.1390438.

Donald Metzler and W. Bruce Croft. Latent concept expansion using markov random fields. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 311–318, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-597-7. doi: http://doi.acm.org/10.1145/1277741.1277796.

Mandar Mitra, Amit Singhal, and Chris Buckley. Improving automatic query expansion. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–214, New York, NY, USA, 1998. ACM. ISBN 1-58113-015-5. doi: http://doi.acm.org/10.1145/290941.290995.

Dan I. Moldovan and Rada Mihalcea. Using wordnet and lexical operators to improve internet searches. *IEEE Internet Computing*, 4(1):34–43, 2000. ISSN 1089-7801. doi: http://dx.doi.org/10.1109/4236.815847.

Hans-Michael Müller, Eimear E Kenny, and Paul W Sternberg. Textpresso: An ontology-based information retrieval and extraction system for biological literature. *PLoS Biol*, 2(11):e309, 09 2004. doi: 10.1371/journal.pbio.0020309.

P Nadkarni, R Chen, and C Brandt. UMLS concept indexing for production databases: a feasibility study. *Journal of the American Medical Informatics Association*, 8(1):80–91, 2001.

Gábor Nagypál. Improving information retrieval effectiveness by using domain knowledge stored in ontologies. In *On the Move to Meaningful Internet Systems 2005: OTM Workshops*, Lecture Notes in Computer Science, pages 780–789, 2005. doi: 10.1007/11575863_98.

Roberto Navigli and Paola Velardi. An analysis of ontology-based query expansion strategies. In *Proceedings of Workshop on Adaptive Text Extraction and Mining (ATEM) in the* 14^{th} *European Conference on Machine Learning (ECML)*, pages 42–49, Cavtat-Dubrovnik, Croatia, 2003.

Jeffery Pound, Ihab F. Ilyas, and Grant Weddell. Expressive and flexible access to web-extracted data: A keyword-based structured query language. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD*, page to appear, 2010.

Ken Q. Pu and Xiaohui Yu. Keyword query cleaning. *Proc. VLDB Endow.*, 1(1):909–920, 2008. doi: 10.1145/1453856.1453955.

Yonggang Qiu and Hans-Peter Frei. Concept based query expansion. In *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 160–169, New York, NY, USA, 1993. ACM. ISBN 0-89791-605-0. doi: http://doi.acm.org/10.1145/160688.160713.

Roy Rada and Ellen Bicknell. Ranking documents with a thesaurus. *JASIS*, 40(5):304–310, 1989.

Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner. Development and application of a metric on semantic nets. *IEEE Transactions on Systems Management and Cybernetics*, 19(1): 17–30, 1989. doi: 10.1109/21.24528.

Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI'95: Proceedings of the 14th international joint conference on Artificial intelligence*, pages 448–453, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

Thomas C. Rindflesch and Alan R. Aronson. Ambiguity resolution while mapping free text to the UMLS metathesaurus. In *Proceedings of Annual Symposium on Computer Application in Medical Care*, pages 240–244, 1994.

Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science, JASIS*, 41(4):288–297, 1990.

Ralf Schenkel, Anja Theobald, and Gerhard Weikum. Semantic similarity search on semistructured data with the xxl search engine. *Information Retrieval*, 8(4):521–545, 2005. ISSN 1386-4564. doi: http://dx.doi.org/10.1007/s10791-005-0746-3.

Erich Schweighofer and Anton Geist. Legal query expansion using ontologies and relevance feedback. In *Proceedings of the 2nd Workshop on Legal Ontologies and Artificial Intelligence Techniques*, pages 149–160, 2007.

Urvi Shah, Tim Finin, Anupam Joshi, R. Scott Cost, and James Matfield. Information retrieval on the semantic web. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 461–468, New York, NY, USA, 2002. ACM. ISBN 1-58113-492-4.

Steven J. Shute and Philip J. Smith. Knowledge-based search tactics. *Information Processing & Management*, 29(1):29 – 45, 1993. ISSN 0306-4573. doi: DOI: 10.1016/0306-4573(93)90021-5.

Anne Sihvonen and Pertti Vakkari. Subject knowledge improves interactive query expansion assisted by a thesaurus. *Journal of Documentation*, 60(6):673 – 690, 2004.

Amanda Spink, Rider I Building, Dietmar Wolfram, and Tefko Saracevic. Searching the web: The public and their queries. *Journal of the American Society for Information Science and Technology*, 52:226–234, 2001.

Padmini Srinivasan. Optimal document-indexing vocabulary for MEDLINE. *Information Processing & Management*, 32(5):503 – 514, 1996a. ISSN 0306-4573. doi: DOI: 10.1016/0306-4573(96)00025-8.

Padmini Srinivasan. Query expansion and MEDLINE. *Inf. Process. Manage.*, 32(4):431–443, 1996b. ISSN 0306-4573. doi: http://dx.doi.org/10.1016/0306-4573(95)00076-3.

Padmini Srinivasan. Retrieval feedback in MEDLINE. *Journal of the American Medical Informatics Association*, 3(2):157–167, 1996c. doi: 10.1136/jamia.1996.96236284.

Nenad Stojanovic. On the query refinement in the ontology-based searching for information. *Information Systems*, 30(7):543–563, 2005. ISSN 0306-4379. doi: http://dx.doi.org/10.1016/j.is.2004.11.004.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: a core of semantic knowledge. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 697–706, New York, NY, USA, 2007. ACM Press. ISBN 9781595936547.

Anja Theobald. An ontology for domain-oriented semantic similarity search on XML data. In *BTW*, pages 217–226, 2003.

Martin Theobald, Holger Bast, Debapriyo Majumdar, Ralf Schenkel, and Gerhard Weikum. TopX: efficient and versatile top-k query processing for semistructured data. *The VLDB Journal The International Journal on Very Large Data Bases*, 17(1):81–115, January 2008.

Tudhope, Douglas, Binding, Ceri, Blocks, Dorothee, Cunliffe, and Daniel. Query expansion via conceptual distance in thesaurus indexed collections. *Journal of Documentation*, 62(4):509–533, 2006. ISSN 0022-0418. doi: 10.1108/00220410610673873. URL `http://dx.doi.org/10.1108/00220410610673873`.

Jouni Tuominen, Tomi Kauppinen, Kim Viljanen, and Eero Hyvönen. Ontology-based query expansion widget for information retrieval. In *Proceedings of the 5th Workshop on Scripting and Development for the Semantic Web (SFSW 2009), 6th European Semantic Web Conference (ESWC 2009)*, May 31 - June 4 2009.

Amos Tversky. Features of similarity. *Psychological Review*, 84:327–352, 1977.

David Vallet, Miriam Fernández, and Pablo Castells. An ontology-based information retrieval model. In *The Semantic Web: Research and Applications, ESWC 2005*, pages 455–470, 2005.

Ellen M. Voorhees. Query expansion using lexical-semantic relations. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 61–69, New York, NY, USA, 1994. Springer-Verlag New York, Inc. ISBN 0-387-19889-X.

Haofen Wang, Yan Liang, Linyun Fu, Gui-Rong Xue, and Yong Yu. Efficient query expansion for advertisement search. In *SIGIR '09: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 51–58, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-483-6. doi: http://doi.acm.org/10.1145/1571941.1571953.

Yih-Chen Wang, James Vandendorpe, and Martha Evens. Relational thesauri in information retrieval. *Journal of the American Society for Information Science*, 36(1):15–27, 1985. ISSN 0002-8231.

Ryen W. White, Ian Ruthven, and Joemon M. Jose. A study of factors affecting the utility of implicit relevance feedback. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 35–42, New York, NY, USA, 2005. ACM. ISBN 1-59593-034-5. doi: http://doi.acm.org/10.1145/1076034.1076044.

Barbara M. Wildemuth. The effects of domain knowledge on search tactic formulation. *J. Am. Soc. Inf. Sci. Technol.*, 55:246–258, February 2004. ISSN 1532-2882.

D. Wollersheim and J. W. Rahayu. Ontology based query expansion framework for use in medical information systems. *International Journal of Web Information Systems*, 1(2), 2005a. ISSN 1744-0084.

Dennis Wollersheim and J. Wenny Rahayu. Using medical test collection relevance judgements to identify ontological relationships useful for query expansion. In *ICDE Workshops*, page 1160, 2005b.

Jinxi Xu and W. Bruce Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM Trans. Inf. Syst.*, 18(1):79–112, 2000. ISSN 1046-8188. doi: http://doi.acm.org/10.1145/333135.333138.

Yiming Yang. Expert network: effective and efficient learning from human decisions in text categorization and retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 13–22, New York, NY, USA, 1994. Springer-Verlag New York, Inc. ISBN 0-387-19889-X.

Yiming Yang and C G Chute. Words or concepts: the features of indexing units and their optimal use in information retrieval. In *Proceedings of the Annual Symposium on Computer Application in Medical Care*, pages 685–689, 1993.