Terminological cleansing for improved information retrieval based on ontological terms

Antonio Jimeno-Yepes yepes@ebi.ac.uk European Bioinformatics Institute Wellcome Trust Genome Campus Hinxton, Cambridge, UK Rafael Berlanga-Llavori berlanga@lsi.uji.es Dept. of Computer Systems and Languages Universitat Jaume I, Spain

Dietrich Rebholz-Schuhmann rebholz@ebi.ac.uk European Bioinformatics Institute Wellcome Trust Genome Campus Hinxton, Cambridge, UK

ABSTRACT

Ontologies are frequently used in information retrieval being their main applications the expansion of queries, semantic indexing of documents and the organization of search results. However, the optimization of an ontology to perform information retrieval tasks is still unclear. In this paper, we propose an ontology query model to analyze the usefulness of ontologies in effectively performing document searches. Moreover, we propose a series of heuristic techniques that optimize ontologies for information retrieval tasks. Preliminary results demonstrate that current domain ontologies provide enough information to support user requests and that ontologies might be improved with simple methods.

1. INTRODUCTION

Ontologies and terminological resources have appeared in information retrieval (IR) either to provide query expansion terms, to perform semantic indexing of documents or to produce a better organization of retrieved documents. However, these ontologies are usually not optimized for IR tasks.

In this paper we rely on language modeling [19], as it provides a formal probabilistic background and an interesting retrieval performance. In language modeling, documents are ranked by the probability that the query has been generated by the language model of the document. In our work, we combine a query model produced by an ontology and the models of the documents. The query model is based on the topology of the ontology and a selection of concepts from the ontology (that represents an information need). This model, combined with the document model, is used to rank the documents. In this paper, we study the relation between

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Copyright 200X ACM X-XXXXX-XX-X/XX/XX ...\$5.00.

ontologies and the retrieval task, providing mechanisms to improve ontologies in a way that these changes provide better retrieval performance. These mechanisms rely on heuristics and relevance feedback.

In the next section, we present related work. Afterwards, the query model is introduced and then compared against state of the art approaches. Results are analyzed and several proposals for ontology refinement are presented. Finally, proposals related to lexicon cleansing are applied to improve the ontology lexicon.

2. RELATED WORK

The contribution of this paper is related to query expansion (QE), as some QE techniques usually rely on knowledge resources such as thesauri and ontologies[10]. However, the main difference between our approach and QE methods is that our queries are generated from ontology concepts instead of free text. Indeed, we propose a novel retrieval model that produces IR queries starting from a set of selected concepts from an ontology. As a consequence, techniques used for selection and weighting of words for expanded queries will be different from those used for generating ontology-based queries. In this section we present related work in QE where the selection of expansion words is based on either a document collection or a knowledge source.

2.1 Collection Dependent

Relevance feedback.

the user runs a query and selects some of the documents that considers relevant [6, 5]. The words are selected from the relevant documents D_r and non-relevant documents D_{nr} marked by the user and are combined using, for instance, Rocchio or Ide. Rocchio defines the expanded query as:

$$\vec{q}_m = \alpha \vec{q}_0 + \beta \frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j$$
 (1)

The vector of the original query \vec{q}_0 is modified according to the vectors of the relevant documents D_r and irrelevant documents D_{nr} . The parameters $(\alpha, \beta \text{ and } \gamma)$ are used

to adjust the relevance of the original query, the vector of the relevant documents and the vector of the non-relevant queries respectively.

Feedback from a user is expensive and usually is not provided. To overcome this problem, pseudo-relevance feedback (or blind feedback) considers the first k documents from the retrieved set as relevant before applying relevance feedback. Even though this approach has shown to be effective, sometimes decreases the retrieval performance. Mitra et al. [17] proposed a strategy to filter the first k retrieved documents before applying pseudo-relevance feedback. A similar relevance feedback method is proposed in [14] for language models, which are called relevance models.

Global strategy.

statistics from a collection are considered. In some cases, co-occurrences are indicators of relevance[22]. One of the parameters in this technique is the window size from which the words are considered to estimate the co-occurrences. Qui and Frei[20] built a statistical thesaurus that is used to select words for query expansion. The problem of the co-occurrences is that it will not find associations between words that do not appear in the same document and that may be semantically related like astronaut and cosmonaut. Latent semantic indexing (LSI)[9] solves this problem. The dimensions obtained using LSI provide an explanation from the mathematical point of view but might be difficult to interpret them semantically. Another issue with the probabilistic model of Latent Semantic Analysis (LSA) is that it assumes that words and documents form a joint Gaussian model, while a Poisson distribution has been observed[21] in information retrieval. More recent models based on a multinomial model reported better results. However, these models are computationally expensive and the update of the index is complex.

Local strategy.

retrieved documents are examined at query time to determine words for expansion. Clustering techniques are used to find clusters from which the words will be selected. As it is applied at retrieval time, the fetching and processing of the documents make it difficult to adapt to on-line systems. A combination of global and local analysis is proposed in [24].

2.2 Knowledge Source Dependent

As mentioned earlier, knowledge sources have been widely used in information retrieval. Voorhees[23] used WordNet and a TREC collection and applied manual query expansion. The combination of the terms is based on the extended vector space model by Fox et al.[11] (convenient to combine multiple sources). She found that query expansion improves the performance for short queries. Longer ones contain more terms providing a better specification of the information need and usually the expansion decreases the retrieval performance. In addition, she noticed that each query has specific peculiarities that make the problem even more difficult. Bodner and Song[4] have shown an improvement based on manual query expansion and general and domain-specific knowledge sources.

Kekäläinen et al.[13] studied the impact of query structure and the query expansion on retrieval performance. They found that with weak structures and Boolean structured queries, QE was not effective. The use of strong structures (more elaborated search key relations) was more effective. Nie et al. [18] integrated logical operators with the vector space model in combination with WordNet. They added expansion terms into the original query as Voorhees, but using fuzzy logic. They showed that adding the expansion terms into the original query does not improve retrieval because the emphasis is moved towards these terms. They proved that the use of the OR operator improves the performance because the terms are alternatives to the original query terms without biasing the user interest on the expansion terms.

In the biomedical domain, Aronson et al.[1] and Liu et al. [15] experimented with Medline using UMLS and the LocusLink database to retrieve documents specific to genes. Chu et al.[8] have applied query expansion using UMLS and the OSHUMED collection based on template specific queries in which the expansion terms are combined based on the relations in the UMLS and the document collection.

2.3 Discussion

Collection dependent approaches might produce dependences among words denoting a conceptual relation that we want to exploit in our work to improve our ontology. Query expansion based on knowledge sources requires an arrangement of the terms according to the query language and the context of the query to produce relevant results. Therefore, an specific optimization is required to be performed to the query expansion mechanism.

Language models have also been combined with knowledge sources as in Cao[7]. In this approach, the topology of WordNet is combined with probabilities estimated from a background collection. However, in our approach we aim at estimating probabilities from our own ontology, so that we can refine them according to the information provided by the feedback mechanism. In the following section we propose a novel ontology-driven retrieval model, which relies on language models and ontologies with rich lexicons. Afterwards, we propose several heuristics aimed at optimizing these ontologies for IR tasks.

3. ONTOLOGY QUERY MODEL

The main aim of the ontology query model (OQM) is to produce an IR query from a set of concepts \mathcal{C} selected by a user browsing the ontology. As we have seen, existing QE methods select and combine terms linked to concepts from an ontology but do not consider their distribution in an ontology. In our approach, we are interested in defining a query model for ranking documents according to the terms appearing in the lexicon of the ontology. Thus, we start from the set of words provided by the lexicon of the ontology, denoted by LexW(T); where T is the set of terms. This means that the term breast cancer in T will be represented as the words breast and cancer in LexW. The terms are grouped in synsets LexT; e.g. breast cancer is placed in the same synset as mammary carcinoma. A given synset is linked to a concept in the ontology. In addition, we require that such a model easily reflects the ontology changes that might improve its retrieval effectiveness.

In the OQM, we need to estimate $P(w_i|\mathcal{C})$, that is, the probability of generating the word w_i given a set of concepts \mathcal{C} . In other words, we want to estimate the probability of choosing the word w_i when expressing the concepts in \mathcal{C} in written documents. On the other hand, each document has

its own language model D, which is estimated by observing the frequencies of the words contained in it. The model of C is then compared to each document model D from the collection using cross-entropy:

$$CE(\mathcal{C}, D) = \sum_{w_i \in LexW(T)} P(w_i|\mathcal{C})log(P(w_i|D))$$
 (2)

In this way, we can obtain the list of ranked documents for the information request expressed by C.

3.1 Estimation of the OQM

This query model is similar to the translation model presented in [3] but applied to the query instead of the document model. We can still apply different estimations of the document model, for instance, relating the different words [7].

The document model is represented by the Jelinek-Mercer smoothed probability of a word in a document and the probability of the word in a background document collection G.

$$P(w_i|D) = \lambda \frac{freq(w_i, D)}{\sum_{w_k \in D} freq(w_k, D)} + (1 - \lambda)P(w_i|G) \quad (3)$$

The relation between the concepts has to be considered in the query model. As the document model is built based on a bag of words instead of multi word terms as the ones linked to the concepts, so the terms have to be linked back to the individual words they are composed of. We propose an implementation of the $P(w_i|\mathcal{C})$ as a smoothed version of the concept model P_{CM} using the expansion $P_R[2]$ based on related concepts:

$$P(w_i|\mathcal{C}) = \lambda_r P_{CM}(w_i|\mathcal{C}) + (1 - \lambda_r) P_R(w_i|\mathcal{C})$$
 (4)

The conceptual model CM considers the probability of the word w_i and the concept C_l and the probability of selecting the concept from C.

$$P_{CM}(w_i|\mathcal{C}) = \sum_{C_l \in C} P(w_i|C_l)P(C_l|\mathcal{C})$$
 (5)

The probability of a word given the concept C_l depends on the probability of the term being related to the concept and the probability of the word belonging to the term (LexT denotes the synset linked to a concept).

$$P(w_i|C_l) = \sum_{t_k \in LexT(C_l)} P(w_i|t_k)P(t_k|C_l)$$
 (6)

Then the probability of the word w_i given the term t_k intends to capture the specificity of the word. Words like *protein*, very frequent in the lexicon of the ontology, will have a low probability.

$$P(w_i|t_k) = \frac{freq(w_i, t_k)}{\sum_{t_m \in LexT(O)} freq(w_i, t_m)}$$
(7)

A smoothed version allows incorporating the probability of the word in the dictionary of the ontology lexicon (W_c) .

$$P_s(w_i|t_k) = \lambda_T P(w_i|t_k) + (1 - \lambda_T) P(w_i|W_c)$$
 (8)

The probability of a given term to be relevant to the concept accepts several implementations and it is left in the

general formulation as the val formula. A simple approximation, used in our work, implements val as the number of concepts linked to the term, where C(O) is the set of concepts in the ontology.

$$P(t_k|C_j) = \frac{val(t_k, C_j)}{\sum_{C_l \in C(O)} val(t_k, C_l)}$$
(9)

A smoothed version is provided

$$P(t_k|C_j) = \lambda_C \frac{P(C_j|t_k)P(t_k)}{P(C_j)} + (1 - \lambda_C)P(t_k|C(O))$$
 (10)

Finally, $P(C_i|\mathcal{C})$. This probability is assumed uniform as shown in the following formula. It may be suppressed since it is constant for all the concepts:

$$P(C_l|\mathcal{C}) = \frac{1}{|\mathcal{C}|} \tag{11}$$

The formulation of the expansion is expressed as a translation probability model from which probability of the words of related concepts are considered. We will follow a different approach compared to Bai et al.[2] since we estimate the probability based on the ontology and the lexicon and not on the documents.

$$P_R(w_i|\mathcal{C}) =$$

$$\sum_{t_k} \sum_{C_n} \sum_{C_l} P(w_i|t_k) P(t_k|C_n) P(C_n|C_l) P(C_l|\mathcal{C})$$
 (12)

Where $t_k \in LexT(C_n)$, $C_n \in C_r(C_l)$ (C_r denotes the related concepts of a given concept) and $C_l \in \mathcal{C}$.

 $P(C_n|C_l)$ provides the probability of picking a concept C_n considering the concept C_l . We express the relation between the concepts using the following formula, where rel has different possible implementations. In our work we have implemented rel as the frequency of the relation in the ontology.

$$P(C_n|C_l) = \frac{rel(C_n, C_l)}{\sum_{C_o \in C(O)} rel(C_n, C_o)}$$
(13)

We can provide a smoothed version (P_s) of the formula where C_r is the set of all the relations in the ontology. $P(C_n|C_r)$ estimates the probability of finding a concept related to another concept, e.g. related species linked to many proteins will not be considered so relevant. In the current formulation we consider taxonomic and other types of relations. More specific probabilities can be assigned in a different formulation if required.

$$P_s(C_n|C_l) = \lambda_r P(C_n|C_l) + (1 - \lambda_r) P(C_n|C_r)$$
 (14)

The model allows different estimation probabilities for the relations. For instance, in the case of genes and disease we are not interested on terms related to the species of the protein since it does not improve the specificity of the query. In addition, another valid approach allows us to consider relevance depending on the type of relation that may be estimated empirically or set in a heuristic manner.

In our approach we rely on the distance between concepts. Distant concepts will have a lower probability for the user to

select them. In addition, this formulation considers concepts that are related to many other concepts. These concepts are not specific to the concepts in the query so they should get a lower weight.

4. PERFORMANCE OF OQM

In this section we describe the experiments carried out to show the effectiveness of queries generated from a domain ontology.

4.1 A Biomedical Ontology

The ontology used for the experiments is the result of merging several well-known biomedical sources. Every source has its own data model and an adaptor is needed to translate them into the target ontology model. We are interested on concept types relevant for gene and disease association and protein-protein interaction, this means: protein, genes, diseases, species, protein function and cellular localization. Thus, data coming from sources are filtered to keep just the information we are interested to include in the ontology. For example, we consider only the disease branch of the MeSH and only some entries from the Gene Ontology Annotation based on the evidence type that supports the fact. We have considered the Swissprot database for the proteins and genes (PGN) and the GPSDB database as a source for PGN names.

The resources are integrated into a common repository with a common representation that reduces the overhead of reusing the ontology. The alignment of the different sources has been done using concept identifiers since some of the data sources are connected to each other by external links. The lexicon has been cleaned of common English terms from a stop word list, ambiguous terms used in a large number of concepts, terms with one character and terms from GPSDB also included in the Gene Ontology (they are not proper protein names). We have prepared the ontology in OWL Lite.

4.2 Datasets

We have used two datasets for our experiments. One considers the role of a gene in a disease and the second the interaction of proteins. These two datasets are presented in turn.

4.2.1 PGN-disease data set

We have used the 2005 TREC Genomics collection because there is an interest on generic topics. This collection is made up of a subcollection of Medline, around 4M documents between years 1999 and 2004, and a collection of 50 queries. The queries in this benchmark are categorized in five groups defined by generic topic templates (GTT) ¹. We have selected a set of queries relating PGNs to diseases. Queries are based on a topical template: the role of gene X in disease Y. The slots X and Y are fixed and are instantiated for each query. From the queries that appear in TREC, we have considered 20 queries related to the topical template. The terms in the slots of the template have been manually mapped to concepts in the ontology.

4.2.2 PPI data set

Protein-protein interaction (PPI) databases rely on experimental data or hand-curated analysis from the literature. Usually these systems rely on the retrieval of documents and on the extraction of relevant information in PPI systems; as an example the BioCreative II 2 . This task is really relevant to the biomedical domain.

We would like to select relevant documents that indicate the relation between two proteins and are useful for curation. The DIP database 3 deals with protein-protein interaction on yeast and has pointers to Medline articles. In this database the yeast species has been more carefully curated than any other species.

In total 260 queries are prepared. These queries are represented by database identifiers defined by Uniprot accession numbers that can be mapped to concepts in our ontology. The average number of relevant documents per query is two. The number of queries is larger than in our previous benchmark, so more significant results might be found. The relevance assessment is done based on documents collected for each one of the interacting proteins. The main difference is that the analysis is done based on the full text of the documents instead of abstracts. This means that some documents have been annotated with many interactions, which indicates that a high-throughput method has been used in the reported experiments, and it is unlikely that relevant information is contained in the abstract. These documents are discarded from the set of relevant documents. The document collection contains Medline citations till September 2004, about 15M Medline documents.

4.3 Retrieval Results

We have used the Lemur package to set up the experiments. A standard stop word list and the Krovetz stemmer are used. The parameters for the different models have been chosen empirically. The lambda values for the ontology query model, presented in section 3, have been set to 0.6. The number of documents used to build a relevance model is 5, more documents provide worst results in both sets. Randomization test for paired data is used to compare the methods[25] († indicates p < 0.10 and ‡ indicates p < 0.01).

In table 1 and 2 we compare the language models (LM) with the relevance models (RM) [14], the OQM without related concepts (On(r=0)) and the OQM considering the closest concepts (On(r=1)). Several λ values for the document model are used that account for differences in document length[16]. As we can see, the query model without relations has the best performance over the other methods, this means that compared to the language models the query ontology model offers a better integration of the synonyms linked the concepts.

Relevance models do not perform as well as expected. This is due to the fact that there is a bias towards one of the concepts of the query. For example, in table 3 we can see the model obtained for the query Alzheimer disease, IDE, where terms associated to Alzheimer disease are much more relevant than those associated to IDE and protein names. In addition, as documents in Medline are short, retrieved documents are easily biased by ambiguous terms. Even so, the performance of OQM is within the mean performance of the systems presented in Genomics TREC 2005.

¹http://ir.ohsu.edu/genomics/2005protocol.html

²http://biocreative.sourceforge.net/bc2ws/index.html

³http://dip.doe-mbi.ucla.edu/

λ	LM	RM	On(r=1)	On(r=0)
0.00	0.0000	0.0000	0.0000	0.0000
0.10	0.0951	0.0890	0.1407	0.1431
0.20	0.1117	0.1013	0.1388	0.1448
0.30	0.1219	0.1100	0.1375	0.1463
0.40	0.1274	0.1133	0.1354	0.1467
0.50	0.1299	0.1160	0.1341	0.1471
0.60	0.1332	0.1182	0.1333	0.1473
0.70	0.1340	0.1206	0.1324	0.1476
0.80	0.1341	0.1220	0.1321	0.1469
0.90	0.1340	0.1238	0.1294	0.1462
1.00	0.0651	0.0000	0.0000	0.0000

Table 1: Mean average precision of the models according to λ (PGN-disease)

λ	LM	RM	On(r=1)	On(r=0)
0.00	0.0000	0.0000	0.0000	0.0000
0.10	0.1479	0.0330	0.1412	0.1514
0.20	0.1552	0.0336	0.1449	0.1542
0.30	0.1589	0.0349	0.1456	0.1573
0.40	0.1592	0.0353	0.1479	0.1589
0.50	0.1592	0.0353	0.1497	0.1610
0.60	0.1571	0.0351	0.1498	0.1614
0.70	0.1560	0.0345	0.1513	0.1633
0.80	0.1553	0.0353	0.1524	0.1631
0.90	0.1536	0.0359	0.1536	0.1641
1.00	0.0491	0.0000	0.0000	0.0077

Table 2: Mean average precision performance of the models according to λ (PPI)

RM	$P(w_i R)$	On(r=0)	$P(w_i \mathcal{C})$
disease	1.91E-05	insulinase	0.0643
alzheimer	4.22E-06	ide	0.0643
ad	3.21E-06	hs.1508	0.0643
gene	1.68E-06	3.4.24.56	0.0321
genetic	2.68E-07	protease	0.0321
ide	2.12E-07	ec	0.0321
study	5.13E-08	insulin	0.0214
enzyme	9.27E-09	degrading	0.0214
insulin	6.14E-10	enzyme	0.0214
brain	3.81E-10	senile	0.0078

Table 3: Models for the query "IDE and Alzheimer disease" (10 top words)

In the PPI set the performance of the ontology is better than the original query. This is due to the fact that the species have several synonyms and that not always the protein name is the best identifier for the protein. As we can see in table 4, the relevance models still have a preference for one concept or a different one that rank non-relevant documents first.

RM	$P(w_i R)$	On(r=0)	$P(w_i \mathcal{C})$
protein	1.79E-8	rad52	0.0889
dna	1.62E-11	yml032c	0.0889
rad51	4.87E-12	cerevisiae	0.0667
rad52	8.04E-14	saccharomyce	0.0667
strand	1.87E-14	yer095w	0.0533
recombination	5.45E-18	mut5	0.0533
repair	2.95E-19	baker	0.0444
rad54	2.82E-23	yeast	0.0444
gene	1.57E-24	homolog	0.0267
stimulate	5.30E-26	reca	0.0267

Table 4: Models for the proteins RAD51_YEAST and RAD52_YEAST (10 top words)

From these results, we can identify several directions to refine the ontology in order to improve the OQM retrieval. These issues are the following ones:

- Inaccuracies in specific terms associated to concepts are not appropriate for retrieval. These terms may be ambiguous and redundant adding noise to the produced query model.
- New terms are needed for existing concepts; some concepts are expressed using different terms, collecting those terms is relevant to reduce the number of false negatives.
- New concepts are needed for existing terms; some terms may be ambiguous but this ambiguity is not reflected in the ontology.
- New concepts with new terms are necessary: concepts related to the query concepts not existing in the ontology.

In the next sections we present several approaches for ontology cleansing and the results compared to the approaches presented here.

5. LEXICON CLEANSING

As already mentioned, sources for false positives are: ambiguous terms, redundant terms never used for a specific concept or terms that do not appear in Medline. These terms do not contribute to increase the recall but to increase the number of false positives that are being retrieved (query drift). Several heuristics are presented in this section.

The first heuristic consists in removing terms from the lexicon that are not found in the document collection (Medline). This heuristic is query independent and allows removing redundant terms and, consequently, reducing space and noise in the lexicon.

The second strategy is aimed at finding the specific contexts in which a concept is labeled with a term. This heuristic is based on either the co-occurrence of terms labeling the query concepts or their related concepts in the ontology.

Finally, the third heuristic consists in using some documents deemed relevant to determine which terms are preferable to label the ontology concepts. This strategy is similar to that adopted by some IR systems, where a small number of relevant documents are used to retrieve more effectively the remaining documents. As the PGN-disease database contains many relevant documents per query, we randomly pick up the relevant documents from the retrieved set. In this way, selected terms are likely to belong to the relevant set. However, the PPI set contains an average of two relevant documents per query and it is not possible to use the previous method. In this case, we have used pseudo-relevance feedback as the baseline.

In the following section we introduce the different heuristics used to select terms to be discarded and finally we compare the performance of these proposals in our two datasets.

5.1 Term Removal Candidate Selection

5.1.1 Terms not in Medline (Corpus)

Some terms never appear in Medline but appear in the lexicon. These terms add noise to the query model giving less relevance to more relevant terms. These terms are sometimes descriptions of the concepts or terms present in specific databases that are not useful for retrieval. We have to remember that due to the length of the Medline documents we expect to have specific terms more than descriptions of the concepts.

We propose to remove from the lexicon the terms that do not appear in Medline or the link between a given term and a concept. Once this heuristic is applied, approximately half of the terms in the lexicon are targeted to be removed.

5.1.2 Query Concept Terms Co-occurrence (QCCoor)

As mentioned above, some terms are used in some contexts to denote a concept. Similar to Cao et al.[7] the cooccurrences between the terms in the corpus can be used as reference to determine this use. Techniques in information retrieval that use the notion of co-occurrence have been introduced in section 2. The algorithm 1 estimates the co-occurrence of each query concept term. The frequency $freq_{t,c_j}$ is estimated querying the information retrieval system where the term t appears labeling the query concept c_i and any term labeling the query concept c_j .

 ${\bf Algorithm~1~Query~Concepts~Terms~Co-occurrence~cleansing}$

```
1: Given concept set C and corpus D
 2: for all c_i in C do
3:
      for all c_i in C do
 4:
        if c_i != c_j then
 5:
           for all term t in LexT(c_i) do
             freq_t + = freq_{t,c_i}
 6:
7:
           end for
 8:
        end if
9:
      end for
10:
      for all term t in synset LexT(c_i) do
11:
        if freq_t < \alpha then
12:
           remove t from synset LexT(c_i)
13:
         end if
14:
      end for
15: end for
```

5.1.3 Co-occurrence of Related Concepts in Medline (QCOnto)

The use of terms to denote concepts might be discovered using the related concepts in the ontology and looking at their co-occurrences in text. The algorithm 2 depicts pseudo-code of the implementation that we have used.

Algorithm 2 Ontology and collection cleansing

```
1: Given Concept c with terms t and relations r and corpus
2: Collect terms from linked to the concept C that appear
   in corpus D
3: for all cr in r do
4:
      for all tr in LexT(cr) do
        for all t do
5:
6:
           if t and tr co-occur then
7:
             freq_t + = freq_{t,c_r}
8:
           end if
g.
        end for
10:
      end for
11: end for
12: for all t do
13:
      if freq_t < \alpha then
14:
        remove t from synset LexT(c)
      end if
15:
```

5.1.4 Relevance algorithm (Rel)

16: end for

Given some relevant documents we can estimate the probability of the terms labeling concepts occur in relevant documents and discard the terms with low probability of being used. During the analysis of the results we identify the set of terms to determine if there is a pattern of term preference based on this approach. We have used the same approach but selecting the documents with blind feedback (RF).

5.2 Lexicon Cleansing Result

The heuristics proposed for lexicon cleaning have a positive effect in retrieval performance. As we see in table 5 4 , the best method is based on the analysis of some of the relevant documents. We have compared the individual queries to better understand the behavior of the approach compared to the baseline ontology query model.

		PGN-disease	PPI
	Baseline	0.1476	0.1641
j	Corpus	0.1635	0.1693
	Coocur	0.1581	0.1736^{\dagger}
	Onto	0.1711^{\dagger}	$\boldsymbol{0.1736^{\dagger}}$
	RF	0.1524	0.1673
	Rel	0.3080^{\ddagger}	_

Table 5: Lexicon cleansing results (MAP)

Concerning the PGN-disease set, in table 6 (see appendix) we find that removed terms are either a description/type of the concept (Acute Confusional Senile Dementia) or identifiers that are never used in text (hs.1508). There are terms that by composition have a very close meaning like mammary carcinoma while the more precise term breast cancer

 $^{^4\}alpha$ has been set to 1 for the Coocur and Onto approach

is used. This is a phenomena that we have already found in[12] with the terms *cancer* and *neoplasm* that usually appear in the same synset. These two terms are close according to their meaning but do not mean the same.

We already mentioned that the cleaning based on relevance has the best performance; therefore it is interesting to derive conclusions from it. Few terms are kept from the terms collected in the synsets. In addition, the number of relevant documents is increased meaning that the removed terms were adding noise.

Compared to baseline methods, relevance feedback does not produce an interesting result. This is related as well to the result obtained with the relevance models. In addition, these phenomena have already been found in the TREC Genomics. The documents might be too short to add terms that are related but do not target the precise terms required for retrieval. On the other hand, the method based on the ontology (Onto) produces an interesting result compared to the other methods since there is no notion of retrieval relevance.

In the PPI set, the average number of documents per query is close to two, so we have not applied the relevance approach presented above. The heuristics applied do not require a notion of relevance.

As we can see in table 7 (see appendix), there are several terms that are targeted for removal. Descriptive terms, e.g. *DNA repair protein RAD51*, identifiers and specific terms without usage in co-occurrence with query concepts or ontology concepts, e.g. *mut5*, are removed.

Relevance feedback has a lower performance than the other methods. We have already observed this behavior in the PGN data set and with the relevance models.

There are less relevant documents retrieved at the cut point of 1000 but looking at the precision at different levels, the missing documents are not found among the first documents as we can find in the different precision results at different ranks. This means that the missed documents will not be found in any case by the user that will prefer to reformulate the query before looking at the complete list.

As there is not much information that is integrated in the ontology for yeast we find that the result with the cooccurrences is similar. On the other hand, we find that the result is quite relevant since without a notion of relevance we can identify and target terms that are not interesting for retrieval.

5.3 Discussion

The approaches used to clean the lexicon improve the performance over the initial ontology. The method relying on relevant documents has the best performance. This method allowed us to reduce the set of terms considerably. This means that query terms need to be very precise and may give clues about the appropriate representation of the query for Medline abstracts. The other methods present similar performance but the one based on co-occurrence of related terms presents a better performance. Blind relevance feedback has a lower performance than the other methods. We have already observed this behavior in the PGN and PPI data set with the relevance models and in TREC Genomics.

The methods presented in this paper allow to remove some of the false positives. There are terms that are more difficult to remove without an indication of relevance that are close in meaning by composition. These results may explain the behavior of the relevance models in Medline. The documents contain very specific words related to concepts of interest. The presence of words that are not so specific like *protein* seem to bias the intention of the query. The behavior in full text articles might be different.

Another interesting result is that due to the method used, all the terms that have not been removed by the algorithm appear in Medline. We can just remove all the terms that are not found in Medline and produce a reduced version of the lexicon. In addition, this has an important technical benefit since half of the documents do not appear in Medline.

6. CONCLUSIONS

The ontology query model presents an interesting performance and we plan to investigate on the different parameters to obtain further improvement in retrieval. We would like to consider, as well, the combination of a query model with different document models.

In this paper we enumerate directions for further ontology revision. We intend to investigate on the relations in the ontology and refine the current set to the most useful ones and to complete the set of relations using information extraction.

In the biomedical articles the information is mostly hypothetical and can be invalidated by posterior research. We plan to investigate on the validity of this data and the implications in the retrieval performance.

7. REFERENCES

- [1] A. R. Aronson and T. C. Rindflesch. Query expansion using the UMLS Metathesaurus. In *Amia*, 1997.
- [2] J. Bai, D. Song, P. Bruza, J. Nie, and G. Cao. Query expansion using term relationships in language models for information retrieval. *Proceedings of the 14th ACM* international conference on Information and knowledge management, pages 688–695, 2005.
- [3] A. Berger and J. Lafferty. Information retrieval as statistical translation. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pages 222–229, 1999.
- [4] R. C. Bodner and F. Song. Knowledge-based approaches to query expansion in information retrieval. In *Canadian Conference on AI*, pages 146–158, 1996.
- [5] C. Buckley and G. Salton. Optimization of relevance feedback weights. In Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, pages 351–357. ACM Press, 1995.
- [6] C. Buckley, G. Salton, J. Allan, and A. Singhal. Automatic query expansion using SMART: TREC 3. In Text REtrieval Conference, 1994.
- [7] G. Cao, J. Nie, and J. Bai. Integrating word relationships into language models. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, pages 298–305, 2005.
- [8] W. Chu, Z. Liu, and W. Mao. Textual document indexing and retrieval via knowledge sources and data mining. In Communication of the Institute of

- Information and Computing Machinery (CHCM), Taiwan, 5(2), 2002.
- [9] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.
- [10] E. Efthimiadis. Query expansion. In Appeared in: Williams, Martha E., ed Annual Review of Information Systems and Technologies (ARIST), v31, pages 121–187, 1996.
- [11] E. Fox. Lexical relations: Enhancing effectiveness of information retrieval systems. SIGIR Forum, 15(3):5–36, 1980.
- [12] A. Jimeno-Yepes, P. Pezik, and D. Rebholz-Schuhmann. Information retrieval and information extraction in tree genomics 2007. In In Proceeding of Ninth Text REtrieval Conference (TREC-15). National Institute of Standards and Technology, Special Publication, 2007.
- [13] J. Kekäläinen and K. Järvelin. The impact of query structure and query expansion on retrieval performance. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, pages 130–137. ACM Press, 1998.
- [14] V. Lavrenko and W. Croft. Relevance based language models. Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, 2001.
- [15] Z. Liu and W. Chu. Knowledge-based query expansion to support scenario-specific retrieval of medical free text. In SAC '05: Proceedings of the 2005 ACM symposium on Applied computing, pages 1076–1083, New York, NY, USA, 2005. ACM Press.
- [16] D. Losada and L. Azzopardi. An analysis on document length retrieval trends in language modeling smoothing. *Information Retrieval*, 11:109–138, 2008.
- [17] M. Mitra, A. Singhal, and C. Buckley. Improving automatic query expansion. In Research and Development in Information Retrieval, pages 206–214, 1998.
- [18] J. Nie and F. Jin. Integrating logical operators in query expansion in vector space model. 8, 2002.
- [19] J. Ponte and W. Croft. A language modeling approach to information retrieval. In SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, pages 275–281, New York, NY, USA, 1998. ACM.
- [20] Y. Qiu and H. Frei. Concept-based query expansion. In Proceedings of SIGIR-93, 16th ACM International Conference on Research and Development in Information Retrieval, pages 160–169, Pittsburgh, US, 1993.
- [21] T. Roelleke. A frequency-based and a poisson-based definition of the probability of being informative. In Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval, pages 227–234. ACM Press, 2003.
- [22] C. J. van Rijsbergen. Information retrieval. Butterworths, London, 2 edition, 1979.
- [23] E. Voorhees. Query expansion using lexical-semantic

- relations. In Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval, pages 61–69. Springer-Verlag New York, Inc., 1994.
- [24] J. Xu and W. Croft. Query expansion using local and global document analysis. In Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 4–11, 1996.
- [25] A. Yeh. More accurate tests for the statistical significance of result differences. In Proceedings of the 18th conference on Computational linguistics, pages 947–953, Morristown, NJ, USA, 2000. Association for Computational Linguistics.

Ontology		Lex. Clean		Coocur		Onto		RF	
rad52	0.0889	rad52	0.0778	rad52	0.0778	rad52	0.0778	rad52	0.0778
saccharomyce	0.0667	cerevisiae	0.0583	saccharomyce	0.0583	saccharomyce	0.0583	cerevisiae	0.0583
cerevisiae	0.0667	saccharomyce	0.0583	cerevisiae	0.0583	cerevisiae	0.0583	saccharomyce	0.0583
mut5	0.0533	mut5	0.0467	rad51	0.0467	rad51	0.0467	rad51	0.0467
baker	0.0444	homolog	0.0233	baker	0.0389	baker	0.0389		
yeast	0.0444	reca	0.0233	yeast	0.0389	yeast	0.0389		
homolog	0.0267	protein	0.0117	homolog	0.0233	homolog	0.0233		
reca	0.0267	rad51	0.0117	reca	0.0233	reca	0.0233		
repair	0.0133	repair	0.0117						
rad51	0.0133								

Table 7: $P(w_i|\mathcal{C})$ for the proteins RAD51_YEAST and RAD52_YEAST (10 top words)

Ontology		Lex. Clean		Coocur		Onto		RF		Rel	
insulinase	0.0643	insulinase	0.0643	insulinase	0.0571	ide	0.0571	insulinase	0.0571	ide	0.0571
ide	0.0643	ide	0.0643	ide	0.0571	insulin	0.0190	ide	0.0571	insulin	0.0190
hs.1508	0.0643	protease	0.0321	protease	0.0286	degrading	0.0190	insulin	0.0190	enzyme	0.0190
3.4.24.56	0.0321	insulin	0.0214	insulin	0.0190	enzyme	0.0190	degrading	0.0190	degrading	0.0190
protease	0.0321	enzyme	0.0214	degrading	0.0190	dementia	0.0069	enzyme	0.0190	disease	0.0046
ec	0.0321	degrading	0.0214	enzyme	0.0190	alzheimer	0.0069	senile	0.0069	alzheimer	0.0046
insulin	0.0214	senile	0.0078	dementia	0.0069	presenile	0.0069	presenile	0.0045		
degrading	0.0214	disease	0.0078	senile	0.0069	disease	0.0069	dementias	0.0045		
enzyme	0.0214	dementia	0.0078	disease	0.0069	onset	0.0035	primary	0.0035		
senile	0.0078	presenile	0.0052	alzheimer	0.0046	last	0.0035	dementia	0.0035		

Table 6: $P(w_i|\mathcal{C})$ for the query "IDE and Alzheimer disease" (10 top words)