UNIVERSITAT JAUME I
DPTO. DE LENGUAJES Y SISTEMAS INFORMÁTICOS


# Ontology Refinement for Improved Information Retrieval in the Biomedical Domain

Ph. D. Thesis

Presented by Antonio José JIMENO YEPES
Supervised by Dr. Rafael BERLANGA LLAVORI

Castellón, APRIL 2009

# Resumen

## Introducción

Las ontologías de dominio están recibiendo la atención de varios campos de investigación ya que permiten codificar nuestro conocimiento del dominio en un formalismo que se puede compartir. Este conocimiento ofrece varias posibilidades que no ofrecen las bases de datos o los lexicones ya que las ontologías son capaces de codificar la semántica del dominio y de proveer mecanismos para hacer inferencia. Sin embargo, el beneficio que las ontologías pueden tener en los Sistemas Informáticos actuales no ha sido definido de forma clara y es difícil estimar la relevancia de las mismas; aunque sea sólo de manera comparativa.

La recuperación de la información ha alcanzado un grado de eficacia aceptable con el uso de los modelos de lenguaje. En el caso general su rendimiento difícilmente se supera mediante otro tipo de técnicas; aunque todavía quede camino para obtener un método con un rendimiento óptimo. La especialización de la búsqueda a un caso de uso concreto podría proveer un resultado más óptimo. En este contexto, resulta interesante optimizar un sistema de recuperación de la información para un tipo de consultas dado dentro de un dominio concreto.

Las ontologías de dominio presentan beneficios potenciales que son difíciles de medir. El poder estimar la correlación entre una ontología de dominio y un caso de uso podría permitir una optimización del desarrollo y refinamiento de dichos recursos que son caros de generar y mantener. Además, el poseer una medida de dicha correlación, aunque sea únicamente para comparar dos ontologías, permitiría que en ciertos casos se pudiese automatizar parte del trabajo ofreciendo la información más relevante para ser considerada para el desarrollo de la ontología. Este trabajo de tesis doctoral se centra en el uso de ontologías de dominio y su refinamiento enfocado a la recuperación de la información.

## Objetivos

En este proyecto de tesis estamos interesados en usar ontologías de dominio para mejorar la recuperación de la información. El dominio seleccionado ha sido el de la Biomedicina ya que dispone de una extensa colección de resúmenes en la

base de datos Medline. Además existen recursos que facilitan la creación de ontologías muy extensas, tales como MeSH o UMLS; aunque la semántica de estos recursos no sea la más adecuada para realizar inferencias. En esta tesis se plantean varios objetivos:

- Aplicación de las ontologías para la recuperación de la información. En este sentido investigamos los modelos de consulta con los modelos de documento que permiten seleccionar los términos relevantes para la consulta y que podrían relacionar la ontología con los modelos de recuperación de la información.

- Estimación de la correlación entre las ontologías de dominio y una tarea concreta que nos permita estudiar las carencias de una ontología y estudiar posibles mejoras.

- Automatización del refinamiento de las ontologías basado en un método que detecte posibles refinamientos tras el análisis de los problemas encontrados para la recuperación de la información.

- Las bases de conocimiento existentes no contienen todos los hechos relevantes para ser usados en el refinamiento. La literatura biomédica contiene esta información pero está presente de forma no estructurada. En este trabajo de tesis investigaremos la combinación de la recuperación de la información y la extracción de la información para el refinamiento de ontologías de dominio a partir de dichas bases de conocimiento.

## Desarrollo

Como hemos comentado, nuestra propuesta de refinamiento de ontologías está basada en un método que considera los problemas de la ontología dentro de la tarea de recuperación de la información en el dominio de la Biomedicina.

La recuperación de la información está basada en modelos de lenguaje y modelos de relevancia. La ontología ha sido integrada en el sistema de recuperación utilizando los modelos de lenguaje. Así, el modelo de los documentos se combina con un modelo de consulta basado en la ontología que se integra fácilmente en la recuperación mediante el cálculo de la entropía cruzada entre el modelo del documento y la consulta. El modelo de consulta tiene en cuenta la topología de la ontología y la distribución de los términos en el léxico enlazado. La ontología se ha desarrollado a partir de recursos existentes, alineando y uniendo varios recursos semánticos.

El método de refinamiento de la ontología enlaza la ontología, la recuperación de la información y la relevancia de los documentos de acuerdo a las consultas generadas. La extracción de la información se combina junto con un sistema de retroalimentación para proveer posibles refinamientos de la ontología.

El sistema implementado para la extracción de la información usa técnicas ya conocidas y aportaciones que hemos implementado para casos específicos

como la anotación de enfermedades y la extracción de relaciones mediante co-ocurrencias y un sistema de filtrado de frases relevantes.

La evaluación de nuestra propuesta requiere la selección de medidas que nos permitan comparar los distintos métodos. En este caso, las medidas estándar de evaluación utilizadas en la recuperación de información han sido las más adecuadas. Durante la evaluación compararemos el comportamiento de métodos existentes en los que se incluirán los métodos tradicionales de retroalimentación (*relevance feedback*) comparados con el uso de la ontología y la ontología refinada. Además de la comparación con métodos existentes, nuestra intención es investigar las distintas condiciones en las que se puede usar nuestra propuesta para el refinamiento de la ontología.

Hemos trabajado con dos conjuntos de datos basados en Medline. Uno de los conjuntos contiene consultas sobre el rol de un gen en una enfermedad obtenidos a partir de Genomic TREC 2005 y el otro conjunto está compuesto de documentos sobre la interacción de proteínas de la levadura basado en anotaciones de la base de datos DIP.

## Resultados

Los resultados muestran que el modelo de consulta basado en la ontología es efectivo para recuperar documentos según una preselección de conceptos por parte del usuario. A partir de este resultado hemos derivado un conjunto de mejoras de la ontología que podrían hacerla más efectiva para la recuperación de la información y hemos usado varias heurísticas para obtener de forma automática la información necesaria para las mejoras.

La limpieza del léxico ha demostrado que los recursos existentes tienen términos que no son usados para denotar los conceptos en ciertos contextos y aproximadamente la mitad de esos términos no aparecen en Medline, lo que permite disponer de un ahorro importante de almacenamiento y una mejora considerable del rendimiento. El mejor resultado se obtiene mediante la presencia de algunos documentos relevantes. Heurísticas basadas en pseudo-relevance feedback no han obtenido buenos resultados. El refinamiento mejora la recuperación de la información si utilizamos métodos que permitan la desambiguación adecuada de los conceptos presentes en los documentos. En este caso, el tener algunos documentos relevantes permite contextualizar mejor los conceptos de la consulta aunque la obtención de términos relacionados basándonos en co-ocurrencias también ha tenido un resultado interesante.

Finalmente, hemos buscado indicadores de la relación de interés en cada uno de los conjuntos de datos y hemos adaptado el modelo de consulta basado en la ontología para contener los términos de las relaciones. Hemos podido comprobar que ante un conjunto de entrenamiento extenso es posible destilar términos relevantes para las relaciones que producen una mejora significativa en la recuperación de los documentos.

## Conclusiones

Los resultados han mostrado que el refinamiento de la ontología aplicado a la recuperación de la información mejora el rendimiento. Como hemos visto, hemos podido identificar información no presente en la ontología que es útil para la recuperación de la información. Además hemos comprobado que el tipo de contenido relevante para las consultas depende de la consulta y está de acuerdo con los resultados existente en el campo de la recuperación de la información. La limpieza realizada al léxico nos ha permitido observar que hay una preferencia por el uso de los términos que es difícil de capturar eficientemente sin una noción de relevancia. No obstante, el uso de heurísticas basadas en la co-ocurrencia de conceptos de la ontología también ofrece resultados interesantes.

La revisión basada en relevancia permite obtener los mejores resultados. El uso de co-ocurrencias no ha sido tan efectivo; aunque más que el pseudo-relevance feedback, debido a la baja precisión obtenida en los primeros documentos recuperados. Los resultados han demostrado que las relaciones contenidas en la ontología no son relevantes para la recuperación de la información y que sólo un conjunto de relaciones permite mejorar la recuperación de la información. Las relaciones más interesantes para el conjunto que trata el rol de un gen en el desarrollo de una enfermedad dada son: las que relacionan el gen o su proteína producto con proteínas con las que interactúa o con enfermedades que están relacionadas con la enfermedad que aparece en la consulta. La relación con las enfermedades relacionadas es diferente entre consultas. Este método trabaja con información que está bien representada en los documentos. Los términos que denotan dicha información ya están presentes en nuestro léxico con lo que no se han detectado nuevos términos relevantes.

Finalmente hemos investigado el tópico de la consulta y si es posible encontrar términos que denoten dicho tópico basándonos en un conjunto de consultas de entrenamiento. Hemos comprobado que con un conjunto de entrenamiento de un tamaño razonable es posible encontrar términos que denoten el tópico. Sin embargo, hemos podido comprobar que en algunos casos, ante la ausencia de datos de entrenamiento de un tamaño razonable no es posible obtener un resultado interesante. Es posible también que haya que realizar una detección de subtópicos, ya que la representación uniforme del tópico no es posible y requiere refinar los distintos subtópicos que permitirían apuntar a cada conjunto de documentos.

## Trabajo Futuro

El modelo de consulta basado en la ontología ha sido preparado para nuestro propósito, pero la propuesta del modelo permite adaptarlo a ciertas tareas de recuperación en las que las condiciones sean distintas. Por ello, en el trabajo futuro será interesante analizar otro tipo de colecciones de dominios diferentes. Uno de los ejemplos se corresponde con la disponibilidad del artículo en vez de sólo los resúmenes. En este caso, se requiere una configuración distinta. Por

otro lado también podría ser interesante la inclusión de los meta-datos asociados a los resúmenes o artículos (p.e. los conceptos MeSH).

El sistema de extracción de la información utilizado cubre parte del dominio. Con la disponibilidad de más datos de entrenamiento podremos entrenar el sistema de extracción para que sea más preciso y con una cobertura más amplia.

El refinamiento de la ontología aprovechará las mejoras del sistema de extracción. El desarrollo de nuevas heurísticas que intenten encontrar información más específica puede beneficiar la búsqueda de datos específicos. Aunque hay que tener en cuenta que la información es siempre hipotética y que es cuestionable su validez; incluso si es información que está afianzada en el dominio. Dicha información puede dejar de ser válida con la aparición de nuevos conocimientos. Aunque existen algunos ejemplos de taxonomías de relaciones, no están lo suficientemente desarrollados como para incluirlos en nuestro trabajo.

Hemos ampliado el modelo de consulta para introducir los términos de las relaciones. En el futuro estaríamos interesados también en desarrollar la detección de tópico y subtópicos e introducirlos en el modelo de consulta. En la literatura ya existen propuestas encaminadas a este tipo de procesamiento, y que pueden servir de base para la extensión del trabajo de esta tesis. Tanto el descubrimiento de subtópicos como el uso de dichos tópicos en recuperación de la información es una línea de investigación abierta y de gran interés para la comunidad.

# Abstract

Ontologies are becoming very popular in several research fields since they encode our domain knowledge into a formal description that can be reused by sharing it. This knowledge offers different possibilities not offered by databases or lexicons due to its semantics and reasoning capabilities. As a consequence, more ontological resources are becoming available. On the other hand, the profit of these ontologies in our daily life has not been clearly defined and it is difficult to estimate the relevance of all these efforts.

Information Retrieval has reached an upper limit with the language models. A specialization of the search mechanism to a certain topic or structured queries would provide additional benefits. In this context, the optimization of a system to deal with some types of queries within a specific domain is of interest. Ontologies could provide the link between the specialized knowledge required and the specialization of the search mechanism.

The Biomedical Domain has received the interest of the text mining community due to the large corpus available through Medline. Even though the content only includes the abstract and not the full text of the citation, a large amount of information is available.

In our work we study the usage of ontologies in Information Retrieval in the Biomedical Domain under the assumption that the knowledge in the documents and the ontology share a common conceptualization. Therefore this knowledge may be used to help the user in document retrieval. The link of the ontologies and text mining is not straightforward and in this work we provide an approach to perform the link based on a lexicon and an ontology. In addition, we develop an ontology query model based on the domain ontology and the lexicon that is combined with the language models.

Ontologies are expensive to build and maintain due to the *knowledge acquisition bottleneck*. We study the possibility of automating different processes in the ontology lifecycle on the basis of feedback provided concerning its performance solving an Information Retrieval Task.

IE is used in order to extract facts from text to be used as another input to the refinement process. Several contributions are done to the Biomedical Information Extraction in terms of disease name entity recognition and resolution and relation extraction.

Our ontology language model has shown better results than the language model approach for the data sets used as gold standard. We have found as well

that for the document retrieval in Medline it is more effective to specify the precise terms than having a query with context terms.

The refinement algorithm analyzes the feedback, either provided by the user or by pseudo-relevance feedback and produces possible changes to the ontology. The algorithm requires to link facts extracted from documents to modifications to the ontology, we have developed a decision process that links the requirements in terms of fact extraction with operations to be performed on the ontology and the lexicon. The operations are analyzed by the refinement algorithm and a decision is done on the operations to be applied to the ontology.

We have studied the annotation of different entity types where our main contribution has been on the annotation of diseases. We have found that simple methods offer a competitive performance compared to more complex methods used in the identification of genes and proteins. This means as well that disease terminology is more standardized. We have developed a system to identify relations between entities that combines co-occurrence and the classification of sentences. We identify relations that are redundant in the collection. A relation between entities may be hypothetical and we have preferred to rely on well established knowledge denoted by statistical means.

The refinement algorithm, again, has performed different refinements on the set comprised by the ontology and the lexicon. The first one is to clean up the lexicon using several heuristics. The lexical entries are collected from different databases and may contain redundant terms or less specific terms than required for the retrieval task. The lexicon cleaning has proved to be effective. It has shown to target specific terms that better denote the concept without ambiguity since lexical entries present in a lexicon may denote senses that are not completely disjoint. We have found as well that many terms in these lexicons never appear as such in the documents.

Then we have analyzed the documents to either add new terms to the lexicon and relate these terms to existing concepts or create a new concept and new relations between the concepts. Different strategies are applied to extract terms from the documents either based on the syntax of the sentence or based on named entity recognition techniques. We have seen that the strategies based on named entity recognition have a better performance since a better normalization of the concepts is done. Selection of terms based on relevance has proved to produce the best results while co-occurrences offer a similar performance and do not require any relevance information denoted explicitly. On the other hand, pseudo-relevance feedback performed poorly in the data set. This result is in tune with the discoveries in the field. This may be due to the low precision at top-n documents, which makes the selected terms to drift the intention of the query. The refinement of terms for the relations produces interesting results if we have enough examples to build a model or prioritize the features. As we have seen for the PPI-data set, we are able to identify terms that are denoting the relations that are effective for retrieval.

**Keywords:** Information Retrieval, Information Extraction, Ontology Refinement, Biomedical Domain

# Acknowledgments

This work would not have been completed without the collaboration of the people that I mention here.

Rafael Berlanga Llavorí for allowing me to be part of his group. In addition, he has done the supervision of the work and helped me through all the process that made possible to present this work. He has been dedicated and focused to identify the problems and to support me during the development of this work. Dietrich Rebholz-Schuhmann gave me the opportunity to work in his research group at EBI on interesting projects which, in a mutual benefit, allowed me to combine the research work for the PhD and my work at EBI. Ernesto Jimenez Ruiz has been my friend and a very important colleague during this work. Part of this work has been the result of collaborating with him. Vivian Lee for the assessment of the disease corpus and Sylvain Gaudan for his statistical approach to named entity resolution. Christoph Grabmüller for proofreading the first version of the manuscript.

Relevant people have participated on the set up of this work even though their role has not been so closely related to this PhD project. Roberto Saban, my boss during my PhD studentship at CERN, supported me financially and sometimes believed in my possibilities more than I did. I would like to mention Bertrand Rousseau for introducing me to search systems and Elena Manola for introducing me to the first application domain in the engineering document system at CERN. A special mention to Professor Pellegrini and Melanie Hilario that allowed me to grow my knowledge in statistical learning even though we did not finish the started project; they made me change my point of view from a pure engineering one to a one focused on research and they introduced me to work in text mining in the Biomedical Domain. Patrick Ruch from whom I learned about Information Retrieval and Natural Language Processing. Gerold Schneider for providing his long dependency parser. Mark Craven for providing access to his data sets.

The PhD study required to complete preparatory courses. I would like to thank Enric Cervera, who opened me the door for preparing the courses remotely. I would like to thank the people related to the Master in Intelligent Systems that allowed me to develop the studies remotely. I would like to thank Arantza Vicente Navarro for arranging the papers to register me as PhD student. She saved me important amount of time and money.

The PhD work is sometimes lonely and you thank for the friendship that

people around you provide. Alejo Bastos Marzal made my time in Geneva easier while I was away from my family and my wife. Jee-Hyub Kim made my time at the University of Geneva easier to take. Piotr Pezik for what we called *therapy coffee* at EBI and his BNC concordancer.

And last but not least I would like to thank my family that had an important role, supporting and encouraging me. I would like to acknowledge specially my parents and my parents-in-law and finally my wife Maika Vicente Navarro for being there, this work would have not be completed without her support and patience.

# Contents

# List of Figures

# List of Tables

# List of Algorithms

# Chapter 1

# Introduction

## 1.1 Motivation

Ontologies are becoming very popular in several research fields since ontologies encode domain knowledge into a formal description that might be reused by sharing it. This knowledge offers different possibilities not offered by databases or lexicons due to its semantics and reasoning capabilities. As a consequence, more ontological resources are becoming available. On the other hand, the profit of these ontologies in our daily life has not been clearly defined and it is difficult to estimate the relevance of all these efforts.

Information Retrieval (IR) has reached an upper limit with the language models. In the general case, the performance is not easily improved with other approaches. On the other hand, IR is still far from optimal performance. A specialization of the search mechanism to a certain topic or structured queries would provide additional benefits since the system can be optimized to solve a specific problem instead of facing a free lunch optimization [164] problem. In this context, the optimization of a system to deal with some types of queries within a specific domain is of interest. Ontologies could provide the link between the specialized knowledge required and the specialization of the search mechanism.

The Biomedical Domain has received the interest of the text mining community due to the large corpora available through Medline, the largest collection of Biomedical citations. In 2008 Medline contained more than 18 million of entries. Even though Medline only includes the abstract and not the full text of the citations, a large amount of information is available.

High throughput techniques allow biologists to analyze a large quantity of data related to proteins, genes and their functions. This data is made available either through well-known resources (e.g. SwissProt, PDB) or through publications; the latter being preferred in many cases. In Figure 1.1 we find statistics that reflect the growth of Medline[1] and EMBL databases [2]. EMBL databases

[1] http://www.nlm.nih.gov/pubs/factsheets/medline.html
[2] http://www.ebi.ac.uk/embl/Services/DBStats

collect information related to Biomedical entities like genes; its growth is still quite large. On the other hand, the documents are the only source for a large part of the available information, and therefore providing efficient access to the Biomedical literature is required.

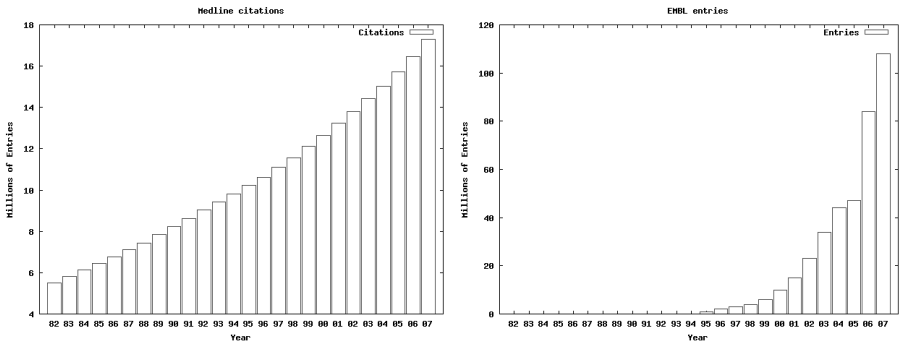


Figure 1.1: Citations in Medline and EMBL per year

## 1.2 Objectives

In our work we study the usage of ontologies in IR under the assumption that the knowledge in the documents and an ontology might share a common conceptualization so this knowledge may be used to help the user in the retrieval of documents. The link between the ontologies and text mining is not straightforward and in this work we provide an approach to perform the link based on both a lexicon and an ontology

The size of current general document collections, like the Web or domain specific like Medline in the Biomedical Domain, increases the different types of information and makes it difficult to find relevant information efficiently. The user may not know all the terms that can be used to express an idea, so some relevant documents are not retrieved, and too many irrelevant documents may be retrieved because the terms are not specific enough to only target the desired documents.

Then, a domain ontology is used as a resource for effective query generation, adding synonyms or related terms, for a topic template query (TTQ). The TTQs are defined by a theme or subject which defines a specific facet of a user information need and by related entities denoted by slots (e.g. the role of gene X in disease Y). Several instantiations of the templates are possible (e.g. the role of the *APC gene* in *colon cancer*).

Ontologies are expensive to build and maintain due to the *knowledge acquisition bottleneck*. There are resources where relevant information is present but the information is not in a structured format (e.g the literature). Semi-automatic or fully automatic methods to analyze the existing unstructured resources are

needed to support the ontology lifecycle. An evaluation methodology for ontologies is a requirement for the automation of ontology learning. We propose to compare two versions of the same ontology based on their performances in IR.

We study the possibility of automating different processes in the ontology lifecycle on the basis of feedback provided concerning its performance solving an IR Task. So, in contrast to current ontology refinement techniques, our method for ontology refinement proposes a solution to the ontology refinement task where the refinement algorithm is driven by the problems detected to retrieve relevant documents. Repairs to these flaws are suggested by applying Information Extraction (IE) to unstructured documents.

Figure 1.2 sketches the modules of the proposed system. The modules process the user conceptual selection in the query formulation module, which uses the ontology and its enclosed lexicon as the source of terms used in the reformulation. The formulated query is taken by the IR module which ranks the Medline documents. The retrieved document set is provided to the user as the answer to the information need.

Our system revises the ontology relying on feedback provided for some queries. A more detailed version of the proposed method is found in Figure 1.3. The IR module communicates to the ontology refinement module the feedback concerning the retrieved documents. Then, a refinement is proposed based on terms extracted from these documents for which a proposal to integrate them in the ontology comes from the IE module. The changes applied to the ontology are reflected in the way the query is generated from the conceptual selection. The idea is that changes applied to the ontology, which might be beneficial for document retrieval, can be used to automate the refinement of the ontology.

This system poses interesting questions that this thesis intends to answer:

- What is the relation between ontologies and text mining? Domain knowledge may be shared among the domain experts and this knowledge may be expressed in the documents in a common conceptualization. But the link between ontologies and text mining tasks is not fully understood[155].
- Is an ontology useful in IR? There is no agreement on the usefulness of techniques like Query Expansion (QE) in IR. Maybe, one of the problems is the specialized knowledge needed for each of the queries; implying that there is no universal procedure in QE for different retrieval scenarios.
- Can we evaluate how well the ontology is able to perform in IR Tasks? The evaluation of the ontology usually is done using a Gold Standard ontology or based on the opinion of domain experts. In our scenario we would like to evaluate the ontology in the IR Task; and use this procedure as a baseline for the evaluation.
- Can we profit from IR feedback to revise the ontology? Theoretical motivation for the refinement algorithm to improve the ontology has to be provided. An analysis of the relevant features to improve retrieval performance has to be done.

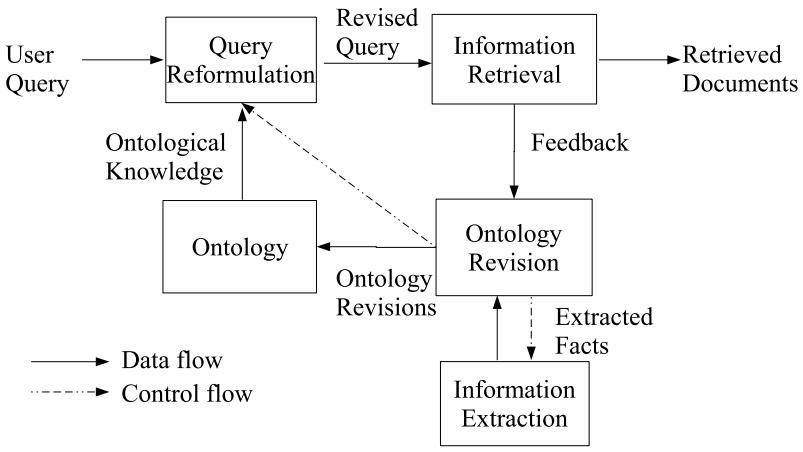


Figure 1.2: System diagram

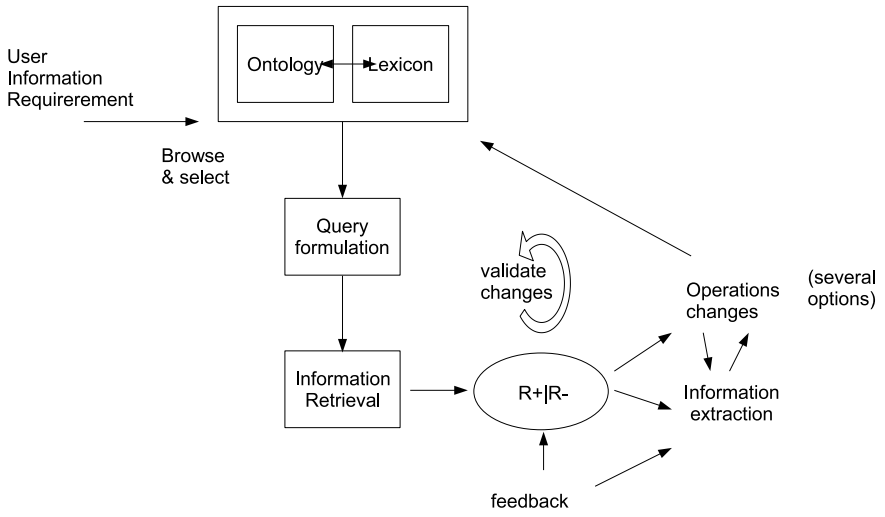


Figure 1.3: Refinement system diagram

- What type of knowledge is discovered by this method? Maybe we are able to improve the retrieval performance, but then, what is the learned knowledge like? Is it really useful for ontology development apart from improving the retrieval task?

- Can we use the facts extracted using IE in the ontology refinement algorithm? The major part of the information is in unstructured sources. A link has to be develop to integrate the extracted information as facts into the refinement procedure.

## 1.3 Organization

The layout of the thesis is split into two main parts. In the first one the different domains are presented. Shortcomings of these domains in view of our work are identified and solutions are proposed. The chapters in this part are:

- Chapter 2 introduces the ontologies and the relations and differences between ontologies and lexicons. We also discuss their relation in the ontology lifecycle and the benefit of combining domain ontologies and lexicons. We discuss the benefits of providing a shared lexicon for the development of ontologies in life sciences. Moreover, we describe the relation between the ontologies and text mining where IR and IE are involved. The benefits of this relation are explored. Then we discuss the evaluation of the ontology given the indirect evaluation on the task the ontology has to fulfill. Finally, we describe our ontology that is a merge of several well known resources in the Biomedical Domain. This ontology will be used in the different experiments presented in this work.

- Chapter 3 introduces background knowledge about IR and the main IR models. Then we present several approaches to reformulate/produce queries based on resources like corpora or knowledge sources. We conclude that no specific query formulation model has been proposed that combines the structure of ontologies and the language models. Then, we introduce our ontology query model, which is combined with the document language model to rank the documents in a collection. This model will be used in the experiments.

- Chapter 4 defines the ontology refinement task and revises existing methods. Then we propose our approach for ontology refinement that uses a decision process based on IE to extract facts from text. The output of this decision process is evaluated for the IR task and the most relevant pieces of information are considered to be included in the ontology.

- Chapter 5 introduces background knowledge about IE. We enumerate the IE needs that were introduced in the decision process for the refinement algorithm and the IE approach used in each task. We have used standard

IE solutions and we propose a post-processing of a co-occurrence analysis to identify related concepts.

The second part presents the experimental setup, the results and finally the conclusions of the work. The chapters in this second section are:

- Chapter 6 describes the typical approach used for evaluating IR systems, this includes benchmarks and measurements. We describe the experimental approach followed in this work and present the data sets used in the experiments. One of the data sets is taken from the TREC Genomics 2005 competition while the second one has been produced based on the DIP database related to yeast protein-protein interactions.

- Chapter 7 presents the results of the proposed method against well-known methods used as baseline. We show that the ontology query model improves the performance of the language models and that it is possible to improve the ontology for IR. We offer different improvements like cleansing of the lexicon linked to the ontology, refinement of the concepts and the relations between concepts and finally we identify terms that are relevant to identify the relation expressed in a TTQ in text.

- Chapter 8 highlights the main conclusions of the work analyzing specific points from the results. Finally future work is proposed that could be seen as the continuation of the experiments to improve the performance or as direction for further related research.

# Chapter 2

# Ontologies

## 2.1 Introduction

The term ontology comes from the Greek words *ontos* (to be) and *logos* (word) an is supposed to study 'what there is'. Gruber [61][62] defines an ontology as a *specification* of a *conceptualization*. A conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose. A specification is the act of naming explicitly. To specify a conceptualization, one needs to state axioms that constrain the possible interpretations for the defined terms.

In the literature, the term ontology is used with a different sense in several domains. Guarino et al. [64] enumerate two possible senses of the term ontology:

- In the philosophical sense an ontology is a particular system of categories accounting for a certain vision of the world. This system does not depend on a particular language.
- In Artificial Intelligence (AI) an ontology is more an engineering artifact that is constituted by a specific vocabulary used to describe a certain reality (an ontology needs a common lexicon [61] so the agents can talk among themselves), plus a set of explicit assumptions regarding the intended meaning of the vocabulary words.

Guarino suggests assigning the term ontology to the AI sense and using the word conceptualization to refer to the philosophical sense. In this chapter we present the relation of ontologies with different available Biomedical/Life Science resources. Nowadays there are many resources that do not present a clear difference between an ontology and a lexicon for instance. Lexicons seem more suitable for text mining since there is a more explicit link between the text and the lexicon rather than to the ontologies. As shown later, we propose a link between the domain ontologies and the lexicons. Even though a large number of resources appear every year, the suitability of these resources to the

problems presented in the Biomedical Domain is an open question. In our work, we investigate the suitability of an ontology to IR presenting approaches that evaluate the ontology for a given task.

This chapter is organized as follows. The following section describes the link between ontologies and text mining and its relevance in query formulation. Then we introduce the ontology lifecycle and identify the refinement algorithm within this lifecycle. We introduce several criteria for the evaluation of ontologies. Afterwards, we enumerate existing ontologies in the Biomedical Domain and other resources, a definition for the ontology and the link between the lexicon and the ontology. Finally, we describe the ontology used in the thesis that merges several existing resources.

## 2.2 Ontologies and Other Resources

The available resources in the Biomedical Domain can be categorized as lexicons, databases or ontologies. Lexicons consist of a compendium of words enriched with information of its usage [73], being concerned with the linguistic properties of words. We may encounter as well the term *terminology*, which is usually referred to as a *specialized lexicon*[17]. Databases rely on the traditional relational model, which provides a strict schema for the instances. The definition of the *concepts* are represented by tables and the relations by foreign keys between tables. This formalism limits the possible formal definitions of the concepts and does not offer many reasoning capabilities; even though approaches exist to store ontologies in databases. Domain ontologies have much more specific purposes than lexicons, as their intended consumers are computer applications rather than humans. Thus, domain ontologies do not need to care about variants and syntactic categories of the terms they use. Ontologies are usually modeled using a representation language. We identify simple formalisms like semantic networks and more complex representation languages that allow to apply inference like frame logic (F-logic) or description logics (DL).

In Figure 2.1 we have ordered the existing formalisms (denoted by boxes) according to their semantic expressiveness. Existing Biomedical resources are placed to their closer formalism. Genuine lexical resources are placed closer to the left of the diagram like the Biolexicon[115], which contains terminology from several resources with some linguistically relevant information. We find as well the UMLS Specialist lexicon that has been used within several NLP and text mining applications. Closer to the limit between a lexicon and an ontology we find several resources that include links between lexical entries (e.g. UNIPROT). More complex resources lie in between the definition of ontology and lexicon like the NCI Metathesaurus, MeSH, SnomedCT and the UMLS Metathesaurus and the OBO ontologies that account for more complex representations similar to semantic networks. Finally, at the end of the spectrum we find more formal ontologies such as Galen[133], which expresses stronger semantics over medical concepts. Unfortunately, these formal ontologies usually lack lexical entries.

Lexical forms present in available resources can be used for labeling ontolog-

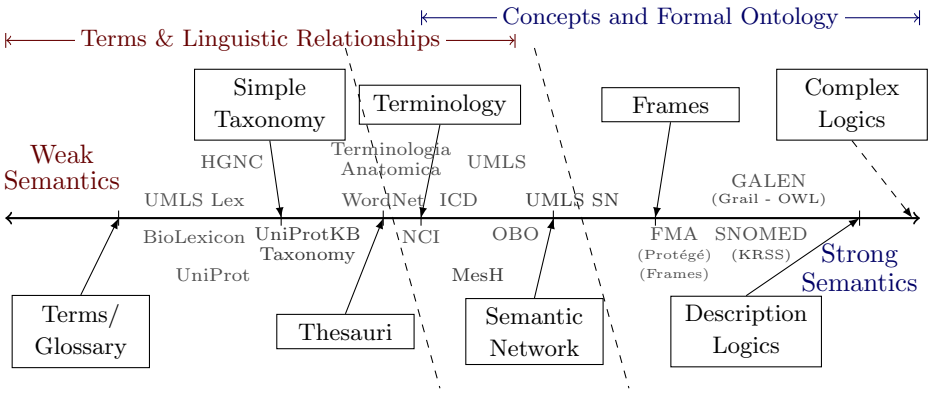


Figure 2.1: Comparative semantic spectrum

ical concepts. The reuse of those labels in different ontologies in combination with a proper definition of the ontological concepts may enable better integration of ontologies.

In the next section we present the relation between ontologies and text mining. We present the problems derived from the phenomena that are typical of the representation of concepts in text and how a lexicon can help as link between the text and the ontology.

## 2.3 Ontologies and Text Mining

Text mining is related to the processing and analysis of textual data. Examples of text mining tasks are text categorization, document retrieval or fact extraction from documents. One of the basic text mining tasks that relate text to ontologies consists of the mapping of concepts in textual sources (e.g. named entity recognition). There are several issues that make this mapping a complex task. The normalization of the concepts present in text, mapping a surface form in text to a concept identifier in an ontology, has to confront different phenomena that are known in natural language. The first one is that the same idea is presented using different surface forms of a given term (morpho-syntactic variations) or by different terms (synonymy). The latter problem is due to the fact that the same term may denote more than one concept (ambiguity); the context in which the term appears can help to further disambiguate the concepts.

The first problem is solved by natural language processing techniques [77] and by including the missing entries. It happens that the resources for the Biomedical Domain may lack the terminology required for doing this mapping since not all the terms denoting the concepts in text are present[12] or that not all the entities of the Biomedical Domain are present in the ontologies. This gap in the resources is solved by adding the missing terminology. This

process can be supported by term extraction tools (presented in more detail in Chapter 5) where the analysis of significantly represented terms in documents is an interesting source to complete the terminological needs of the resources.

The ambiguity problem has different solutions in the literature. Some ambiguous cases are identified directly from the lexicon [115] while others may consider contextual information present in the documents and finally the topology of the ontology[148]; this reinforces the requirements of having a lexicon in combination with an ontology.

Similar representations have already appeared in the literature as we will see later. But still, it might be possible to link ontologies with documents using a lexicon which makes the link. This link approaches the ontologies to text mining algorithms where the link is done through the representation of these concepts in text.

Then, it is relevant to see when it could be useful to apply this ontology and lexicon learning from text. The following section presents the ontology lifecycle and shows different approaches based on text that have been used to support its development.

A further consideration is required since information present in the scientific literature is, in many cases, still hypothetical[155]. Results from experiments produce hypothesis about drug-gene relations or gene-diseases that sometimes require further experimentation or that are refined in a later stage. This means that the knowledge learned from these resources is hypothetical and subject to be increased or falsified. In our work we assume a monotonic increase of knowledge avoiding specific information for which there is not enough evidence from text.

## 2.4 Ontology Lifecycle

We are interested in ontology refinement but this is only one of the steps in the ontology lifecycle. This section covers the steps in the ontology lifecycle from the first analysis to the final refinement of the ontology following the schema presented in[103]. Other proposals for the lifecycle exist but we want to consider this approach, which is closer to learning from text as we will see in the different steps of the lifecycle.

### 2.4.1 Import and Reuse

In this step, the first objective is to define the layout of the ontology based on discussions with domain experts and existing data sources like databases or other ontologies. Due to the increasing number of knowledge sources in the Biomedical Domain, an interest to put together all these complementary sources is needed; the automation of this task is more than necessary. The combination of different sources may be trivial if an explicit link between concepts exists, similar to a foreign key in relational databases. Unfortunately this is not the case and semi-automatic techniques based on semantic distances have appeared

in the literature[63]; Stoilos et al.[151] have shown several techniques combined with text mining. These techniques have been used in the Biomedical Domain [56] and specifically the Anatomical Domain [170]. Special care is needed while providing these links since the ontologies should share the same conceptualization.

Depending on how the ontologies are combined, two major techniques appear as defined by Noy et al.[112], ontology alignment and ontology merging. In ontology alignment, the two ontologies remain untouched and links between them are created. This technique is used when ontologies deal with complementary domains. Thompson et al.[154] created the Mao system to align nucleic acid and protein sequences. Rosse et al.[136] used the system OBR to integrate domain ontologies in anatomy, physiology and pathology and Smith and Rosse[147] worked with the with the Foundational Model of Anatomy (FMA). In ontology merging, the two ontologies are merged into a single one. This technique is used when it is necessary to provide a consistent and unified ontology.

### 2.4.2 Ontology Learning

As we have seen in the introduction, only a small part of the information is available in a structured representation, the remaining part exists in an unstructured representation of which a large portion is textual data. These unstructured sources contain valuable information that might be exploited by data mining algorithms to create or extend an ontology. Several techniques exist that allow for the specific extraction of components that might be integrated in an ontology.

#### 2.4.2.1 Term and concept extraction

Even though this section is described in detail in Chapter 5 several techniques are used to identify new terms/concepts from text that have been mentioned for ontology learning. Jacquemin[77] used a term extractor and the inner term structure (head dependency) to extract a taxonomy of terms that, in addition, are linked by meronymy. Navigli et al.[108] used the system OntoLearn to specialize WordNet for a given domain. Statistical techniques have been proposed too[103].

#### 2.4.2.2 Hierarchical clustering

Hierarchical clustering can be used to group terms based on the context. The resulting structures can be verified by an ontology engineer or domain expert. Blaschke et al.[16] used hierarchical clustering to group gene-products (proteins) from the literature. The result was a set of unrelated trees that were merged by a human curator. Faure et al.[49, 48] used verb and preposition patterns in the ASSIUM system. Not only is the taxonomy of an ontology built, but as well the different verbs that usually co-occur with a given concept. Caraballo [26] uses co-occurrence of appositives and conjunctions within the document collection in

order to build a tree in a bottom-up fashion. A collection of taxonomy learning techniques are presented in Chapter 5.

### 2.4.3 Ontology Pruning

The steps of ontology development presented above collect information and may require pruning the content of the ontology to keep the information that better fits the domain problem. We have to keep in mind the balance between completeness, that has problems in managing the content and the complexity of the processing, and scarcity, that limits the expressiveness of the ontology. Khan[90] proposes an algorithm that has as objective to prune WordNet to avoid the information overhead while looking for information. It is based on a self-organizing tree algorithm called SOTA.

### 2.4.4 Ontology Refinement

The techniques in the import and reuse step have used the data sources massively to build or populate the ontology. Ontology refinement is in charge of the fine tuning of the ontology, different techniques based on IE have already been considered in the literature. Ontology refinement is a very relevant concept in our work and is presented in detail in Chapter 4.

## 2.5 Ontology Evaluation

There are several criteria for evaluating ontologies that study the coherence of the ontology. On the other hand, in ontology extraction the focus is on evaluating the extracted and integrated information against a gold standard used to evaluate an ontology build by manual or automatic means, an evaluation of the classification of existing instances or task-based evaluation and human based evaluation given some predefined standards. An enumeration of different types of evaluations can be found in Brank et al.[19].

Our system considers a task-based evaluation[119]. We are interested in evaluating the utility of an ontology for a given task and then being able to revise the ontology to optimize the performance of the system. Task-based evaluation allows us to evaluate two ontologies according to a given task so we will be able to compare the refined ontology to the original ontology. In our problem the ontology is used as a source for the reformulation of terms so there is a clear link between the ontology and the IR task that is provided by the query formulation mechanism.

## 2.6 Towards a Thesaurus for Life Sciences

We have presented the relation of the lexicon and the ontology, their relation in text mining and how they relate in their lifecycle. A lexicon is relevant during the knowledge acquisition step since the matching of ontological concepts with

existing resources may make easier the reuse of existing concepts, e.g. ontology matching, or the detection of missing concepts that require to be created.

An example is represented by current efforts in the Gene Ontology Consortium that consist of the mapping of existing resources like the MGI database[1] to the Gene Ontology (GO). In specific cases, this mapping can help to detect missing concepts in GO. This mapping still requires validation that is expensive. The existence of a common thesaurus would automatically link the resources and would identify the missing concepts by identifying those not defined in GO but defined in the other resources.

During the different stages of the lifecycle, the lexicon will provide the terminology for existing concepts. If there is no entry in the thesaurus, the current process may suggest the creation of this new entry. The selection of the terms for this entry requires the use of appropriate terms. These terms may be provided by a community effort where several domain experts study the appropriate set of terms and/or using natural language processing (NLP) and text mining[149] to extract terms from the literature[53]. Tools are available to find the terms in context to verify their use, e.g. Keyword in Context Concordancers (KICC)[92].

As a consequence, the existence of a common thesaurus can help to map concepts from existing resources and ensure that we do not recreate concepts and would cluster words by ideas. More specialized knowledge bases like databases and ontologies can link to this common thesaurus. This thesaurus will collect the different terms in a common repository allowing ontologies to be linked accordingly. The idea consists of having one thesaurus and many ontologies. These ontologies may be produced according to different criteria, so no common ontology may be used in different scenarios or use cases. We find the best example in the OBO ontologies where several ontologies can overlap in some of their concepts but the ontologies are not linked or related and several efforts are done separately.

The generation of a common thesaurus requires the resolution of several issues like a common conceptualization linked to the entries in the lexicon. The outcome of the research in the field may require not only creating new concepts, but also to split existing ones. This process will invalidate the link of the current concepts in the ontologies. One way of solving the problem would consist of the generation of several versions.

Although current approaches represent an important initiative for the construction of a shared lexicon they still lack some important requirements to allow a straight forward interoperability with ontologies and text resources. In the next chapters we discuss requirements for the lexicon and text mining tasks (IR and IE) and the results chapter will allow us to further understand the requirements of the lexicon and the ontology and will help to understand better their relation. In the next section we present a more detailed enumeration of existing resources.

[1] http://www.informatics.jax.org/

## 2.7 Biomedical Ontologies, Databases and Thesauri

In the Biomedical Domain[17, 146, 168], there is an important effort to create lexicons, databases and ontologies. Some of the resources are introduced above. The resources presented in this section are of interest for ontology development.

Biomedical ontologies define the concepts of the domain and their relation. Amongst these efforts we find the Open Biomedical Ontologies (OBO) site[2]. These ontologies have been modeled using the OBO language similar to OWL-Lite. The Gene Ontology (GO)[3] is a well known ontology in OBO that has already been used for several text mining tasks[105]. Other efforts include the UMLS Semantic Network, which defines a hierarchy of concepts and relations which covers the medical domain and is used to type the entries from the UMLS Metathesaurus. Anatomical ontologies: Foundational Model of Anatomy (FMA) [4], Adult Mouse Anatomical Dictionary (MA) [5]. Galen and OpenGalen conceptualize entities related to anatomy, surgical deeds and diseases, and the set of relations and modifiers between them.

The output of the experiments might be collected in structured sources like databases. In the Biomedical Domain these databases are concerned about proteins, genes and their function and their relations with other entities like diseases. Protein databases like the Uniprot database (Trembl and Swissprot), the PRINTS database, or the LocusLink[6]. The GPSDB database[117] is a protein synonym database obtained from several protein databases. Other databases deal with different types of entities and relations like the BIND database, the iPRoClass[7] database or the Online Mendelian Inheritance in Man (OMIM)[8].

Terminological resources have been developed to cope with the large domain terminology . The UMLS Metathesaurus[9] from the National Library of Medicine (NLM) is one of the most well known thesauri in the Biomedical Domain. It has been used in several text mining tasks. The Medical Subject Headings(MeSH)[10] is used in addition to annotate Medline. The Systematized Nomenclature of Medicine (SNOMED)[11] contains an extensive collection of medical terminology, around 400.000 clinical concepts. The NCI Thesaurus (NCI) [12] provides terminology on nearly 10,000 cancers and related diseases, 8,000 single agents and combination therapies among others related to cancer and Biomedical research. Finally, the BioLexicon[115] was developed for text mining tasks and collects several relevant *semantic types* that will be enriched with terms extracted from

[2] http://obo.sourceforge.net/
[3] http://www.geneontology.org
[4] http://fma.biostr.washington.edu
[5] http://www.informatics.jax.org/searches/anatdict_form.shtml
[6] http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=gene
[7] http://pir.georgetown.edu/iproclass/
[8] http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM
[9] http://www.nlm.nih.gov/research/umls/
[10] http://www.nlm.nih.gov/mesh/
[11] http://www.snomed.org/
[12] http://cancer.gov/cancerinfo/terminologyresources

text.

## 2.8 Our Biomedical Ontology for Information Retrieval

Our system uses an ontology as source for the query formulation that we denominate Biomedical Ontology for Information Retrieval (BOIR). BOIR ontology is intended to be improved by our ontology refinement algorithm. Our ontology has been designed to cover different aspects of the Biomedical Domain and is based on existing well known data sources. There is more than one formal definition of ontology [65] like Description Logics (DL) (combination for first order logic and frame logic) or F-logics. For an insight on the differences, Corcho et al.[34] make a comparison of representational and reasoning capabilities of some of these languages.

The Semantic web has integrated the description logics and the XML technology. The most popular of these languages is OWL (Ontology Web Language). Our ontology will be expressed using description logics and the ontology will be represented using OWL Lite [13]. Efforts exist to convert existing resources to OWL DL[145] which consider as well the OBO ontologies. The lexicon will be stored in synsets (sets of synonyms) and a link is done between ontology concepts and a synset with the terms that label it.

In the following section we introduce the upper layer of our biomedical ontology which we have used in our query formulation approach. This ontology will be revised by the refinement algorithm.

### 2.8.1 BOIR Model

The expressivity of the ontology is comparable to OWL-lite and is closer to a taxonomy of concepts with some basic relations than more formal ontologies as Galen.

#### 2.8.1.1 Concepts

The types of concepts that our ontology models are:

**Proteins and Genes:** proteins are the product of the genes. They are kept together because the explicit definition of these two related entities is blurred in knowledge bases like Uniprot and in the documents where, in addition, the same term is used to present proteins and genes from different species [155].

**Species:** a group of related organisms that share a more or less distinctive form and are capable of interbreeding. As defined by Ernst Mayr, species are

[13] http://www.w3.org/TR/owl-guide/

groups of actually or potentially interbreeding natural populations which are reproductively isolated from other such groups. [14]

**Molecular Function:** the functions of a gene product are the jobs that it does or the "abilities" that it has. These may include transporting things around, binding to things, holding things together and changing one thing into another. This is different from the biological processes the gene product is involved in, which involve more than one activity. [15]

**Biological Process:** is a recognized series of events or molecular functions. A biological process is not equivalent to a pathway, although some GO terms do describe pathways. Mutant phenotypes often reflect disruptions in biological processes. [16]

**Cellular Component:** locations, at the levels of subcellular structures and macromolecular complexes. Examples of cellular components include nuclear inner membrane, with the synonym inner envelope, and the ubiquitin ligase complex, with several subtypes of these complexes represented. [17]

**Diseases:** any abnormal condition of the body or mind that causes discomfort, dysfunction, or distress to the person affected or those in contact with the person. Sometimes the term is used broadly to include injuries, disabilities, syndromes, symptoms, deviant behaviors, and atypical variations of structure and function, while in other contexts these may be considered distinguishable categories.

#### 2.8.1.2 Relations

Despite the taxonomic relations in the Biomedical Domain we may find interesting relations that allow us to do a fine-grain selection of entries. The data sources do not cover many of the relations and different efforts exist to model this knowledge. We have considered here some of the relations already existing in the data sources that are defined in the UMLS Semantic Network [18]. These relations can be further refined with the appropriate knowledge. In our work we will work with high level relations that can be further refined later on.

**Part_of(cellular component, cellular component)** : composition relation that is applied to subcellular locations, obtained from the Gene Ontology.

**Located_in(protein, cellular component)** : protein and a subcellular location. Gene ontology and the GOA annotation.

**Associated_species(protein, species)** : specifies in which species a protein has been found. Swiss-prot.

[14] http://en.wikipedia.org/wiki/Species
[15] http://www.geneontology.org/GO.function.guidelines.shtml
[16] http://www.geneontology.org/GO.process.guidelines.shtml
[17] http://www.geneontology.org/GO.component.guidelines.shtml
[18] http://semanticnetwork.nlm.nih.gov

**Associated_function(protein, molecular function)** : relates a protein and a function. The information is obtained from the Gene Ontology and the Gene Ontology Annotation.

**Causes_disease(protein, disease)** : relates a protein and a disease. This may be found split among different databases like OMIM. Study how to integrate this with the ontology and how can this help on the ontology refinement.

**protein_protein_interaction(protein, protein)** : relates two proteins that interact. Several databases are available providing this information like DIP, BIND and IntAct.

### 2.8.2 Ontology Engineering

Our ontology is a merge of several sources. Every source has its own data model and an adaptor is needed to translate into the target ontology model defined above. Not only the data model has to be adapted but the information has to be filtered from the different sources. For instance, we consider only the disease branch of the MeSH and only some entries from the Gene Ontology Annotation based on the evidence type that support the fact. This has the advantage that the different resources are accessible from a common repository with a common representation which reduces the overhead of accessing the information. The alignment of the different sources has been done using concept identifiers since some of the data sources are connected by external links. Even though the UMLS already provides a merge of different knowledge sources, it does not provide a clear categorization of terms[19] and does not cover the genes and proteins at the level that we require.

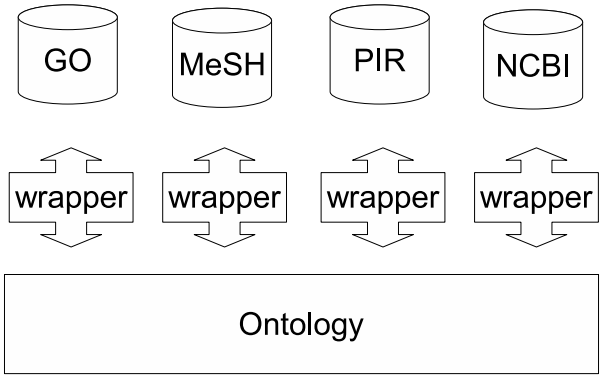


Figure 2.2: Ontology Extraction and Integration

[19]as in prostate cancer for instance, where more than one concept is related to the same term

The list of databases/ontologies/thesaurus used in this work that are imported to our ontology are:

**Gene Ontology (GO)** [20] provides a controlled vocabulary to describe gene and gene product attributes in any organism. Its taxonomy has three main branches (molecular function, biological process and subcellular localization). It includes as well information about the part_of and located_in relations and the related_to_function relation.

**Swiss-Prot** [21] curated protein sequence database. The SwissProt database is part of the Uniprot database that contains Trembl as well. The PGNs defined by Swissprot will be the main entities since almost the other data sources make reference to them.

**Gene Ontology Annotation (GOA)** [22] provides a link between the GO and Swiss-Prot. Filtered by evidence keeping only the ones supported by a curator. This means that for a given protein we may know the location of the protein in the cell, the biological process in which it participates and the molecular function.

**NCBI species taxonomy** [23] contains the species and the taxonomy of the species. In addition, we have the relation between the species and the proteins so we can have a more fine grained distinction of protein depending on the species.

**GPSDB** is a collection of protein synonyms[117] from 14 different protein databases. The adequacy of this terminology for QE will be explored in Chapter 7. The combination of different sources for the identification of proteins in text has shown to be quite relevant to increase the recall as can be found in the BioCreative Gene Normalization task. Some cleaning of the lexicon is required since the different databases collect terminology that may not be appropriate for text mining due to its ambiguity. We can find a common technique to avoid using terms with length less than 3 characters and remove terms that are single numbers. Common English terms (from a stop word list) are deleted and terms that are the name of a family are deleted too. Terms indicating the function of the protein, like tumor suppressor, which are found in the GO are included in the terms to be deleted. And terms that have the same name as the disease they may cause. In the following sections we explain further techniques that have been used to clean the terminology.

[20] http://www.geneontology.org
[21] http://us.expasy.org/sprot
[22] http://www.ebi.ac.uk/GOA/
[23] http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Taxonomy

**MeSH (Medical Subject Headings)** [24] is a controlled vocabulary designed by the National Library of Medicine (NLM) to search Medline and other health sciences databases. From this vocabulary we extracted only the disease taxonomy and integrated it in the ontology. The main concern is that there is no link between the protein and diseases; that can be partially found in some other databases like OMIM; which covers only the inheritable diseases or the NCI gene-disease database.

**PIR protein taxonomy** [25] is a large but incomplete taxonomy of proteins obtained by homology which subsumes them into family and domain. This provides information about the domain, subdomain of the protein. These knowledge sources do not contain a complete definition of the domain; so the application of our query reformulation approach for IR may work better if these sources are expanded.

#### 2.8.2.1 Lexicon cleansing

The terms in the lexicon, collected from different resources, may present different levels of polysemy which may result in different levels of ambiguity in text. This has been well studied in the case of proteins and genes (PGNs). Chen et al. [28] studied the sources of ambiguities for protein names. Jimeno et al. [84] have studied the ambiguity of disease terms in the UMLS. Pezik et al.[115] propose a set of features from the term repository that potentially identify ambiguous terms from the lexicon. We have done a basic cleaning of the lexicon based on common English terms identified in Wordnet. Terms labeling more than a large number of concepts are suppressed. Then protein/gene terminology has been cleaned comparing the terminology with other semantic types like molecular process; this approach has been considered in BioCreAtIvE II by some of the participants. Disease terminology has been cleaned based on a procedure described in [82, 84]; more details are present in Chapter 5 when we describe the disease annotation.

#### 2.8.2.2 Statistics about the ontology

In Table 2.1 we can find the concepts and the statistics. As we can see, the most populated concepts are the proteins and the species. The concept types with more terms are *disease* and *PGN*. The first one belongs to the MeSH, but we can find that the entry terms represent all the possible variants, including plurals and do not represent a set of redundant terms and acronyms appear with their long form. The PGNs always have a large number of entities that which express a high ambiguity level.

In Table 2.2 we find the relation types stored in the ontology and the number of relations that they contain. As we can see, the largest number is represented

[24] http://www.nlm.nih.gov/mesh/meshhome.html
[25] http://pir.georgetown.edu/pirwww/dbinfo/dbinfo.html

| id | name | concepts | terms | terms_concept |
|---|---|---|---|---|
| 1 | molecular_function | 6947 | 9676 | 1.39 |
| 2 | cellular_component | 1254 | 1712 | 1.37 |
| 3 | Protein/genes | 155153 | 951799 | 6.13 |
| 4 | biological_process | 9263 | 12326 | 1.33 |
| 5 | Species | 85934 | 83210 | 0.97 |
| 6 | Diseases | 4164 | 39613 | 9.51 |
| 7 | Analytical, Diagnostic | 2047 | 16610 | 8.11 |
| 8 | Biological sciences | 1563 | 9322 | 5.96 |
| 9 | Organisms | 3395 | 7916 | 2.33 |

Table 2.1: Concept statistics

by the link between the proteins/genes and the species. This is due to the fact that they come from Uniprot where all the proteins are linked to their species.

| id | name | count |
|---|---|---|
| 1 | part_of | 13,852 |
| 2 | has_species | 155,153 |
| 3 | has_function | 3,457 |
| 4 | has_location | 2,549 |
| 5 | has_disease | 3,164 |
| 6 | ppi | 260 |

Table 2.2: Relation statistics

## 2.9 Discussion

In this chapter we have defined the ontologies and the differences and relations with existing resources. Instead of integrating the lexicon into the ontology we have proposed a loose coupling[83]. This coupling is used in the integration of ontologies and text mining. We have proposed the mechanism to evaluate the combination of the ontology and the lexicon for a given task and in the following chapters we will show results in IR. Then we have presented the available resources and the domain that they cover. Based on these resources we have prepared our domain ontology that will be used during the experiments.

In the following chapters we will integrate the ontology into IR, will prepare a mechanism to profit from the feedback provided by the retrieval task into ontology refinement and will integrate this mechanism into IE.

# Chapter 3

# Ontology-based Information Retrieval

## 3.1 Introduction

IR deals with the recovery of documents from a collection for a given user information need expressed with a query. Figure 3.1 shows the typical schema of an IR system. The input to the system is a collection of documents and a query. The output is a set of documents (ranked or not ranked) that matches the criterion for being retrieved. Feedback concerning the retrieval performance can be provided in order to improve the system's behavior; as in relevance feedback. Ad-hoc retrieval is different to text categorization where the user need is fixed and the stream of documents is not fixed. We deal with ad-hoc retrieval but applied to queries defined for a given template (TTQ); as introduced in Chapter 6.

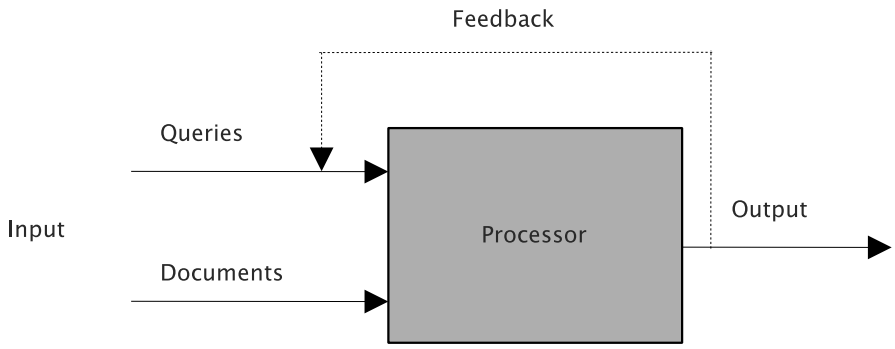


Figure 3.1: IR System

In an IR system, the documents are pre-processed to speed up the retrieval. This means: the documents are tokenized, these tokens are normalized usually turning them into their lower case form and stemmed using available stemmers (e.g. Porter, Levin or Krovetz). In some cases the normalization might decrease the retrieval performance. Special care is required to select the appropriate normalization. In addition, a list of stop words is used to filter these tokens. These words do not carry any meaning and add only noise (e.g. prepositions). Standard stop word lists are available for many languages.

This processing turns the documents into a representation called *bag of words*, that despite its name is a set of words. The outcome of all this processing is a dictionary of terms and a table which for each document has a link to the terms in the dictionary. The query follows the same process prior to retrieval. In Figure 3.2 we can see the tokenization and normalization of the two following sample documents:

**D1: Inhibition of apoptosis by Heliothis virescens ascovirus.**

**D2: Role of apoptosis in biology and pathology.**

In the retrieval process the tokens in the query are searched in the documents' table. This process is very expensive since it requires traversing the whole document collection for each query. To improve the speed, the documents are preprocessed and an *inverted index* is built. Efficient trie structures like the suffix tree are used to locate the terms in the index efficiently. This inverted index is represented in Figure 3.2 where a query containing the term *apoptosis* is linked to D1 and D2.

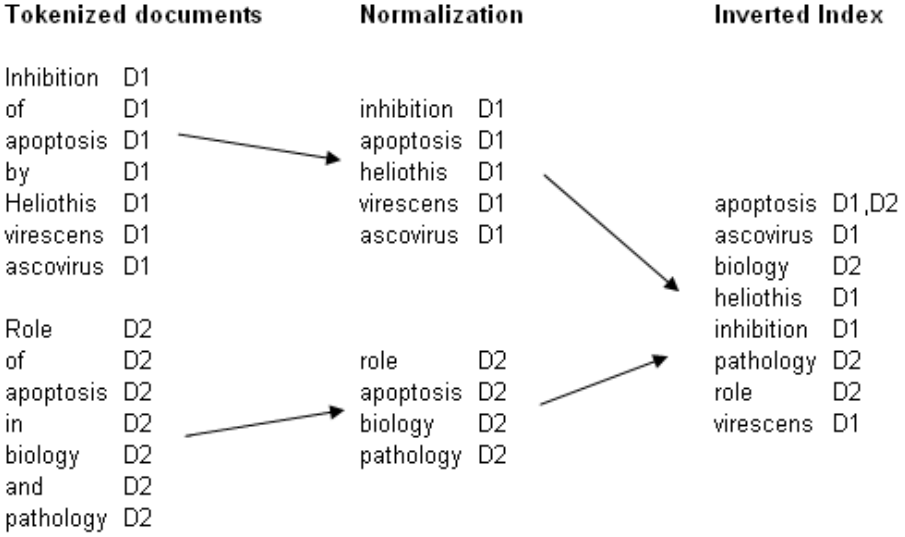


Figure 3.2: Inverted index

This chapter is organized as follows. In the next section we introduce the most popular IR models. This includes the traditional IR models, Boolean, vector space and the probabilistic models and then the language models. Then, we introduce several approaches for enhancing the original user query providing terms from different sources like explicit user feedback or knowledge sources. Finally we present our approach for query enhancement that integrates an ontology and the language models.

## 3.2 IR Models

In this section we present the existing models to perform IR that have a different philosophy concerning the retrieval problem. We will focus on the traditional models and the language models. The models presented in the following sections have in common the usage of an inverted index and will differ in the information stored in their index and in the processing of the user query.

### 3.2.1 Boolean Model

In the Boolean model a document is retrieved if the expression of the query is evaluated as *true*. The most commonly used Boolean operators are AND, OR, NOT. Grouping of keywords using parenthesis allows specifying the order in which the Boolean operators are applied and allows writing more complex queries. Queries can easily be answered using an inverted index since each keyword is linked to a set of documents. An example is presented in the following query:

**Q: apoptosis AND pathology**

The query is decomposed in the keywords K1: apoptosis, K2: pathology connected by the logical operator AND. If we consider our inverted index, K1 is linked to documents D1 and D2 while K2 is linked to the document D2. If we apply then the AND operator on the set of documents returned by K1 and K2 the user will obtain as answer to the query the document D2.

Set operations are usually applied to solve the Boolean operators. The AND operator would perform an intersection, the OR operator would perform a union while the NOT operator will consider the documents that are not in the set defined by its argument. Boolean queries turn out to be complex in some scenarios.

The Boolean operators may retrieve either too many documents or too few documents. Another issue with the Boolean model is that retrieved documents are not ranked according to relevance, even though mixed approaches exist like the extended vector space model or have been formalized in the language models (e.g. Indri[1]).

Boolean systems are still very popular in well known systems like PubMed, this may be because the user understands the process used when documents are retrieved. Almost all the available systems allow the usage of wildcard characters that allow partial matching of the terms in the dictionary.

[1] http://www.lemurproject.org/indri

### 3.2.2 Vector Space Model

In this model documents are represented as vectors in a high dimensional space where each term is a dimension in the index. Terms from the documents are weighted according to statistics like the frequency of the term in the document $f_{t,d}$ and the number of documents in which the term appears $df_t$. The idea behind document frequency is that terms that are very frequent in the collection are less specific; this is expressed with the inverse document frequency (idf):

$$idf_t = log\left(\frac{N}{df_t}\right) \tag{3.1}$$

Each component in the vector combines the estimation of the term frequency and the inverted document frequency.

$$\vec{d} = [w_{1,d}, \ldots, w_{n,d}]^T \tag{3.2}$$

$$w_{t,d} = tf_{t,d} * idf_t \tag{3.3}$$

Documents are compared in this high dimensional space using the cosine of the angle between the vectors of the query and the documents in the collection.

$$cos(\vec{q}, \vec{d}) = \frac{\vec{q} \cdot \vec{d}}{|\vec{q}||\vec{d}|} \tag{3.4}$$

An example is presented in Figure 3.3. The query used in the previous example is used here. The query is converted into the vector representation being the dimensions *apoptosis* and *pathology* the ones having any value. In the figure we can see represented the documents and the query. The documents in this case have been ported from a higher dimensional space just for the example and are an approximation of their distribution in space. As we can see, the representation in vector space of D2 is represented in d2 and is closer to the query vector q than the representation of the document D1 d1. This means that document D2 will be ranked above document D1.

The SMART system[2] has been for some time the most representative system implementing this model. Three letters referring to term frequency, inverted document frequency and the normalization being used underlined in table 3.1 are used to configure the SMART system. The configuration of the retrieval system is defined by the three letters of the document vector and the three letters of the query vector.

There are several well known issues with the vector space model. With the cosine normalization long documents have higher retrieval probability than short documents. Singhal et al.[143] proposed the pivoted cosine normalization

[2] ftp://ftp.cs.cornell.edu/pub/smart

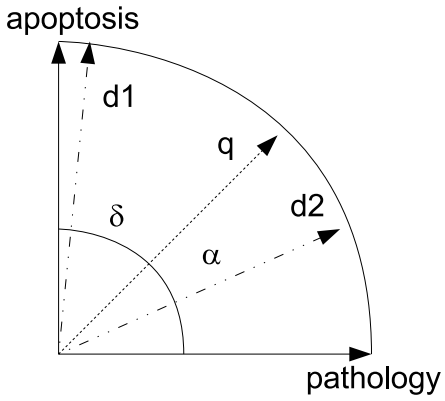


Figure 3.3: Vector space model example

| $tf_{t,d}$ | | $idf_t$ | | norm | |
|---|---|---|---|---|---|
| none | $tf_{t,d}$ | none | 1 | none | 1 |
| binary | 1 | tfidf | $\log(\frac{N}{df_t})$ | sum | $\sum_{i=1}^{n} tf_{i,d}$ |
| max_norm | $\frac{tf_{t,d}}{max\_tf_d}$ | prob | $\log(\frac{N-df_t}{df_t})$ | cosine | $\sqrt{\sum_{i=1}^{n} tf_{i,d}^2}$ |
| aug_norm | $0.5 + 0.5 * \frac{tf_{t,d}}{max\_tf_d}$ | freq | $\frac{1}{df_t}$ | fourth | $\sum_{i=1}^{n} tf_{i,d}^4$ |
| square | $tf_{t,d}^2$ | squared | $\log(\frac{N}{df_t})^2$ | max | $\max tf_{i,d}$ |
| log | $1 + \log(tf_{t,d})$ | | | | |

Table 3.1: SMART weighting schemes

which relates better the length of the document and the probability of being relevant. In addition, the matrix that represents the vector space is very sparse and different solutions have been provided like the latent semantic indexing (LSI)[42] that reduces the dimension of the index and may identify possible relations between the indexing units and the documents that can be used to solve precision or recall issues.

### 3.2.3 Probabilistic Model

The probabilistic model usually is based on the probabilistic ranking principle by van Rijsbergen[157] where documents are ranked according to the probability of relevance based on the information need and is represented by the following formula:

$$P(R = 1|d, q) \tag{3.5}$$

Documents are ranked by decreasing probability of relevance. If a decision has to be taken about the delivery of a document by the system, the Bayes optimal decision rule is applied:

$$P(R = 1|d, q) > P(R = 0|d, q) \tag{3.6}$$

Several assumptions are usually done in the probabilistic model. One of them is the binary independence model, this means that the presence of a term is represented by $x_t = 1$ and the absence by $x_t = 0$. In addition, term order in the document is not considered allowing the simplification of the probability estimation and the independence between documents is achieved.

$$O(R|\vec{d}, \vec{q}) = \frac{P(R = 1|\vec{d}, \vec{q})}{P(R = 0|\vec{d}, \vec{q})} \tag{3.7}$$

Applying Naive Bayes:

$$\frac{P(R = 1|\vec{d}, \vec{q})}{P(R = 0|\vec{d}, \vec{q})} = \frac{\frac{P(R=1|\vec{q})P(\vec{d}|R=1,\vec{q})}{P(\vec{d}|\vec{q})}}{\frac{P(R=0|\vec{q})P(\vec{d}|R=0,\vec{q})}{P(\vec{d}|\vec{q})}} = \frac{P(R = 1|\vec{q})}{P(R = 0|\vec{q})} \cdot \frac{P(\vec{d}|R = 1, \vec{q})}{P(\vec{d}|R = 0, \vec{q})} \tag{3.8}$$

The left part on the right of the equality is query dependent and for ranking purposes can be ignored. Then, Naive Bayes assumption of conditional independence is considered.

$$\frac{P(\vec{d}|R = 1, \vec{q})}{P(\vec{d}|R = 0, \vec{q})} = \prod_{t=1}^{M} \frac{P(d_t|R = 1, \vec{q})}{P(d_t|R = 0, \vec{q})} \tag{3.9}$$

If we consider $p_t = P(x_t = 1|R = 1, \vec{q})$ and $u_t = P(x_t = 1|R = 0, \vec{q})$, the Table 3.2 can help to simplify the estimation of the probabilities.

| | document | relevant(R = 1) | non-relevant (R = 0) |
|---|---|---|---|
| term present | $x_t = 1$ | $p_t$ | $u_t$ |
| term absent | $x_t = 0$ | $1 - p_t$ | $1 - u_t$ |

Table 3.2: Contingency table

$$\frac{P(\vec{d}|R=1,\vec{q})}{P(\vec{d}|R=0,\vec{q})} = \prod_{t:x_t=q_t=1} \frac{p_t}{u_t} \cdot \prod_{t:x_t=q_t=0} \frac{1-p_t}{1-u_t} \tag{3.10}$$

$$\frac{P(\vec{d}|R=1,\vec{q})}{P(\vec{d}|R=0,\vec{q})} = \prod_{t:x_t=q_t=1} \frac{p_t(1-u_t)}{u_t(1-p_t)} \cdot \prod_{t:q_t=1} \frac{1-p_t}{1-u_t} \tag{3.11}$$

Since the right product is query dependent we obtain:

$$RSV_d = log \prod_{t:x_t=q_t=1} \frac{p_t(1-u_t)}{u_t(1-p_t)} = \sum_{t:q_t=1} log \frac{p_t(1-u_t)}{u_t(1-p_t)} \tag{3.12}$$

Then we just need to estimate $u_t$ and $p_t$. If we consider that the relevant documents are a small portion of the collection, the documents that are not relevant can be approximated with statistics over the whole collection. So the $u_t$ is approximated to $df_t/N$ and somehow this is a justification for the inverse document frequency in the vector space model. $p_t$ is related to relevant documents and is related to the frequency of term occurrence in known relevant documents, but we cannot have an explicit enumeration of relevant documents. Relevance feedback, explained later, could be considered as source of relevant documents. Croft and Harper[38] assume a constant value for $p_t$ and propose $p_t = 0.5$, this means $p_t$ and $(1 - p_t)$ cancel each other. So the RSV will be determined by the occurrence of the term in the document and the idf approximation of $u_t$. Greiff[60] proposes to use statistics of $t$ in the collection, this means $p_t = df_t/N$. The most representative system of the probabilistic approach is the Okapy system [131].

### 3.2.4 Language Model

The philosophy of the language model is to rank the documents by the probability that the query has been generated by the language model of the document. A similar idea was already being used in speech recognition and its usage was proposed by Ponte and Croft[118] for IR.

Instead of considering $P(R = 1|q, d)$ as relevance function, the language model approach builds a model for each document in the collection $M_d$. The documents are ranked according to the probability of being generated by the language model of the document $M_d$; $P(q|M_d)$.

In IR the most common formulation is the query likelihood model in which:

$$P(d|q) = P(q|d)P(d)/P(q) \tag{3.13}$$

$P(d)$ is the probability of the document that usually is considered to be the same for all the documents; so it is not useful for document ranking and is discarded. $P(q)$ is the probability of the query so it only depends on the query and it can be discarded too. This means that the documents can be ranked by the probability $P(q|d)$. Typically the unigram model

$$P_{uni}(t_1t_2t_3t_4) = P(t_1)P(t_2)P(t_3)P(t_4) \tag{3.14}$$

is used and has already shown a good performance compared to higher order n-grams where the difference in performance does not seem to provide a significant increase while the resources in terms of indexing space and time are significantly larger. The formulation based on the maximum likelihood estimation (MLE) can be approximated by:

$$\hat{P}(q|M_d) = \prod_{t \in q} \hat{P}_{mle}(t|M_d) = \prod_{t \in q} \frac{tf_{t,d}}{L_d} \tag{3.15}$$

Under the model presented, a document has a non-zero probability if all the query terms appear in the document. Documents containing some but not all of the query terms may still be relevant since the estimation of the models is affected by the sparseness of the data. The solution presented smooths the language model of the document with a reference model that usually is the document collection. This is expressed by the following formula where $t$ is the term and $C$ is the collection, $df_t$ is the number of documents in the collection where the term $t$ appears and $N$ is the total number of documents.

$$P(t|C) = \frac{df_t}{N} \tag{3.16}$$

There are several ways in which the language model is smoothed but two approaches are most widely used, the Jelinek-Mercer smoothing and Dirichlet smoothing. Jelinek-Mercer is defined by a linear interpolation of the language model and the probability of the term in the collection using the $\lambda$ parameter.

$$\hat{P}(t|d) = \lambda \hat{P}_{mle}(t|d) + (1 - \lambda)\hat{P}(t|C) \tag{3.17}$$

Dirichlet smoothing combines the $\mu$ parameter with the length of the document ($tf_{t,d}$) being less sensible to long and short documents[101]:

$$\hat{P}(t|d) = \frac{tf_{t,d} + \mu p(t|C)}{\sum_t tf_{t,d} + \mu} \tag{3.18}$$

Different approaches are applied to estimate the smoothing parameters and the selection of the parameters might be different according to each individual query. This is motivated by the fact that smoothing has different behavior according to the length of the queries and the documents [101].

Hiemstra and Vrie [72] have related the language model to the traditional information retrieval models like the vector space model. Even though experimental work has shown that the language models outperform the vector space

model and the probabilistic model, sometimes a properly trained traditional model performs similar to a language model.

## 3.3 Query Reformulation

A user facing an IR system has to consider how to turn the information need into the system's query language in a way that is effective in terms of retrieval performance. This means turning this need into concepts and from concepts to terms. The user has to consider the different terms that might be used to express the information need in the collection to retrieve only the documents that are of interest. In the Biomedical Domain, biologists may find difficult to keep up to date with all the terminology used in the relevant bibliography, considering the different ways their ideas are expressed and how this term may refer to other existing concepts.

Query reformulation refers to the different operations that are applied to the original user query in order to improve its performance. The advantage of query reformulation is that it is performed on the query side, being possible to reuse the existing indexes and profit from the different lexicons and ontologies to select the terms.

We have further split query reformulation into Query Expansion (QE) and Query Refinement. QE is intended to increase recall and QR is intended to increase precision. The following sections introduce these two approaches.

### 3.3.1 Query Expansion

In domains like the Biomedical Domain where a protein can be specified in the documents using different terms, it is difficult for the user to find all the documents in a document collection. Efthimiadis[46] defines QE as:

> *the process of supplementing the original query with additional terms, and it can be considered as a method for improving retrieval performance. The method itself is applicable to any situation irrespective of the retrieval technique(s) used. The initial query (as provided by the user) may be an inadequate or incomplete representation of the user's information need, either in itself or in relation to the representation of ideas in documents.*

This problem has been tackled from the very beginning of IR using thesauri and statistical techniques. The different techniques can be classified according to the data source for the expansion terms and how they are combined. Based on the source for the terms the techniques depend on the document collection or on knowledge sources:

#### 3.3.1.1 Collection dependent

**Relevance feedback[21, 20]** The user runs a query and selects some of the documents that considers as relevant. The terms are selected from the relevant

and non-relevant documents marked by the user and are combined using Rocchio or Ide. The following formula represents Rocchio's QE:

$$\vec{q}_m = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j \tag{3.19}$$

The vector of the original query $\vec{q}_0$ is modified according to the vectors of the relevant documents $D_r$ and the not-relevant documents $D_{nr}$. The parameters ($\alpha$, $\beta$ 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 documents respectively.

Explicit feedback from the user is expensive and usually the user does not provide it. To overcome this problem, pseudo-relevance feedback (or blind feedback) technique considers the first top-n documents from the retrieved set as relevant and applies relevance feedback. Even though this approach has shown to be effective, sometimes decreases the retrieval performance. Mitra et al. [107] used filtering based on Boolean operators to improve the precision of the top-n retrieved documents before applying pseudo-relevance feedback. Another problem with relevance feedback is that the relevant documents may belong to different clusters[142] while relevance feedback provides only a prototype vector[86]. The language models have a similar approach that uses pseudo-relevance feedback to estimate a relevance model[96]. The relevance models are explained in more detail in Chapter 7 since they are used as one of the baseline methods.

**Global strategy** Statistics from a collection are considered. In some cases, co-occurrences are indicators of relevance[157]. One of the parameters in this technique is the window size from which the terms are considered to estimate the co-occurrences. Qui et al.[122] build a statistical thesaurus that is used to select terms for QE. The problem of co-occurrences is that it will not find associations between terms that do not appear in the same document and that may be semantically related; like *astronaut* and *cosmonaut*. Latent semantic indexing (LSI)[42] solves this problem. The resulting dimensions obtained using LSI provide an explanation from the mathematical point of view but it might be difficult to interpreter them semantically. Another issue is that the probabilistic model of Latent Semantic Analysis (LSA) assumes that words and documents form a joint Gaussian model, while a Poisson distribution has been observed[132] in IR. More recent models based on a multinomial model report better results. These models require high computational power and the update of the index is not trivial.

**Local strategy** Documents retrieved for a query $q$ are examined at query time to determine terms for expansion. Clustering techniques are used to find the clusters from which the terms will be selected. As it is applied at retrieval time, the fetching and processing of the documents make it difficult to adapt it to online systems. A combination of global and local analysis is proposed [166].

#### 3.3.1.2 Knowledge sources dependent

In the literature the selection of terms from a knowledge source is done either manually or automatically. In the manual approach, the user decides which terms are chosen and how they are combined, while in the automatic selection approach the system decides which terms are expanded and how they are going to be combined. Considering the manual approach, Voorhees[158] used WordNet and a TREC collection and applied manual QE. The combination of the terms is based on the extended vector space model by Fox[52], convenient to combine multiple sources. She found that QE improves the performance for short queries. In longer ones QE has less impact since these queries contain more terms providing a better specification of the information need. In addition, she noticed that each query has specific peculiarities that make the problem even more difficult. Bodner and Song[18] have shown an improvement based on manual QE and general and domain-specific knowledge sources.

Kekäläinen and Järvelin [89] studied the impact of the query structure and QE on retrieval performance. They found that with weak structures and Boolean structured queries, QE was not effective. The usage of strong structures (more elaborated search key relations) was more effective. Nie et al. [111] 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 terms into the original query does not improve retrieval because the emphasis is set on the expanded terms. Using the OR operator they proved that there is an improvement because the terms are alternatives to the original query terms without biasing the user interest on the expanded terms.

In the Biomedical Domain, Aronson and Rindflesch[9] and Liu and Chu [100] experimented with the Medline collection using UMLS and the LocusLink database to retrieve documents specific to genes. Chu et al.[29] have applied QE using the UMLS and the OSHUMED collection based on template specific queries in which the expanded terms are combined based on the relations in the UMLS and the document collection (similar to local context techniques). The Genomics TREC track provides a document collection in which different expansion techniques have been used based on LocusLink and the disease branch of the Mesh (Aronson et al.). Systems like PubMed provide automatic QE as a normal practice. It expands synonyms and more specific terms from the MeSH taxonomy as is shown in Table 3.3.

| Original query: Colon cancer |
|---|
| Expanded query:<br>("colonic neoplasms"[TIAB] NOT Medline[SB]) OR "colonic neoplasms"[MeSH Terms] OR colon cancer[Text Word] |

Table 3.3: Query colon cancer expanded by PubMed

Considering automatic QE approach, very accurate word sense disambiguation is needed to target the concepts that appear in the query [59][109][140]. In

the automatic approach, the combination of the original query with the expansion terms depends on the query language (e.g. Boolean operators). In the case of Boolean operators, with exception of the extended vector space model based on Boolean operators, no weight is needed. In the case of the vector space model, a co-occurrence measure like the mutual information is used. This is linked as well with query translation in cross-lingual IR and similar approaches are used where the query is mapped to concepts in a specific multi-language resource, like the UMLS, and then based on the concepts that are translated to the target language[137]. In addition, the queries used to be short and a lack of context may avoid the possibility of disambiguating the query properly.

### 3.3.2 Query Refinement

Document collections like Medline or the Web may easily retrieve a considerable number of documents due to the ambiguity of the query terms. There is an interest not only in increasing recall but as well in increasing precision to reduce the *information overload*. This overload may come because the initial user query is not specific enough and there are different possible interpretations for it. For instance, if you are interested in documents about *APC* in Medline, the system may retrieve documents concerning the gene *adenomatous polyposis coli* or the *anaphase-promoting complex*. In these cases, the user query is not specific enough regarding the documents in the collection. In query refinement we will modify the query in order to filter out irrelevant documents to increase precision. The different techniques look for the appropriate interpretation of the user query considering the documents in the collection. But term ambiguity is not always the only reason for false positives. In some cases it may happen that the document is not really of interest for the intention of the query where the concept is not the main concept in the query and may cover more general issues or too specific.

In most of the cases, the different approaches make suggestions instead of choosing one of the senses but do not always provide satisfactory results. Similar techniques used in IR include Scatter/Gather[39, 69, 70] where the retrieved documents are clustered and a representative label is given to each cluster. The user, based on the label, can visit the clusters that are refined information needs. Pratt et al. [121, 120] work on the classification of the user queries and the clustering of the documents. In the Biomedical Domain we can find the SOPHIA system [113].

Gauch and Smith[57] used an expert system to reformulate the query in an interactive system but marginal results were obtained. Systems relying on an ontology like OntoRefiner (Safar et al.[139]) based on a Galois Lattice study the retrieved documents to display them along the lattice.

Several studies rely on building a user model based on the information extracted from the user session instead of the document collection. To determine the changes from the original query to a refined query, Amati and Bruza [6] used belief change and the query log. This can be seen as building a user profile more than information filtering, where there is only one information need and

the documents come from a continuous stream rather than from a, more or less, fixed document collection.

### 3.3.3 Discussion

We have presented the different approaches to reuse existing knowledge sources and the document collection for query reformulation. We have seen that collection dependent methods work on assumptions on the distribution of terms or on explicit user feedback. The methods depending on knowledge sources have to deal with ambiguity in the query and the combination of terms in the query is language dependent and the final formulation of the query may depend on the retrieval need.

In our problem we require a model that works with our conceptual selection that is integrated with ontologies and the language models. In the next section we present an approach to provide a query model to be integrated with a document model for IR.

## 3.4 Ontology Query Model

In this section, we present an approach to estimate a query model based on an initial selection of concepts from an ontology where the words are provided by the ontology lexicon $LexW(T)$. 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 $\mathcal{C}$ is then compared to each document model $D$ from the collection using cross-entropy:

$$CE(\mathcal{C}, D) = \sum_{w_i \in LexW(T)} P(w_i|\mathcal{C}) log(P(w_i|D)) \tag{3.20}$$

In this way, we can obtain the list of ranked documents for the information request expressed by $\mathcal{C}$.

### 3.4.1 Estimation of the OQM

This query model is similar to a translation model[13] 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[25]. 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 $G$.

$$P(w_i|D) = \lambda \frac{freq(w_i, D)}{\sum_{w_k \in D} freq(w_k, D)} + (1 - \lambda) P(w_i|G) \tag{3.21}$$

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 $P_R$[10] based on related concepts:

$$P(w_i|\mathcal{C}) = \lambda P_{CM}(w_i|\mathcal{C}) + (1 - \lambda) P_R(w_i|\mathcal{C}) \tag{3.22}$$

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 $\mathcal{C}$.

$$P_{CM}(w_i|\mathcal{C}) = \sum_{C_l \in C} P(w_i|C_l) P(C_l|\mathcal{C}) \tag{3.23}$$

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.

$$P(w_i|C_l) = \sum_{t_k \in LexT(C_l)} P(w_i|t_k) P(t_k|C_l) \tag{3.24}$$

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

$$P(w_i|t_k) = \frac{freq(w_i, t_k)}{\sum_{t_m \in LexT(O)} freq(w_i, t_m)} \tag{3.25}$$

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

$$P_s(w_i|t_k) = \lambda_T P(w_i|t_k) + (1 - \lambda_T) P(w_i|W_c) \tag{3.26}$$

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 implements *val* as the number of concepts linked to the term, where $C(O)$ is the set of concepts in the ontology.

$$P(t_k|C_j) = \frac{val(t_k, C_j)}{\sum_{C_l \in C(O)} val(t_k, C_l)} \tag{3.27}$$

A smoothed version is provided.

$$P(t_k|C_j) = \lambda_C \frac{P(C_j|t_k)P(t_k)}{P(C_j)} + (1 - \lambda_C)P(t_k|C_c) \tag{3.28}$$

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:

$$P(C_l|\mathcal{C}) = \frac{1}{|\mathcal{C}|} \tag{3.29}$$

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.[10] since we estimate the probability based on the ontology and the lexicon and not on the documents.

$$P_R(w_i|\mathcal{C}) = \sum_{t_k} \sum_{C_n} \sum_{C_l} P(w_i|t_k)P(t_k|C_n)P(C_n|C_l)P(C_l|\mathcal{C}) \tag{3.30}$$

Where $t_k \in LexT(C_n)$ ($LexT$ denotes the synset linked to a concept), $C_n \in C_r(C_l)$ ($C_r$ 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:

$$P(C_n|C_l) = \frac{rel(C_n, C_l)}{\sum_{C_o \in C(O)} rel(C_n, C_o)} \tag{3.31}$$

We can provide a smoothed version ($P_s$) of the formula where $C_r$ is the set of all the relations in the ontology. In the current formulation we consider taxonomic and other types of relations. More specific probabilities can be assigned in a different formulation if required.

$$P_s(C_n|C_l) = \lambda_r P(C_n|C_l) + (1 - \lambda_r)P(C_n|C_r) \tag{3.32}$$

The radius defines if we select related concepts that are closer or distant to the query concepts. In this formulation we do not make any distinction between taxonomic relations or any other type of relation. For instance, in the case of genes and disease we might not be interested on the terms related to the species of the protein since it does not improve the specificity of the query. In addition,

another valid approach can allow us to consider relevance depending on the type of relation, which may be estimated empirically or set in a heuristic manner.

In our approach we rely on the distance between concepts, so 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 as not specific to the concepts in the query so they should get a lower weight.

The collection of query words from the initial set of concepts may become costly. There are several factors like the cost of selecting the terms from the concepts and from the relations. The computational cost derived by the calculations related to collection frequencies is easily turned into a search in an inverted index being the cost of searching in this index. The cost of the relations is directly related to the radius selected since the order of the polynomial is directly related to the radius $r$ ($O(n^r)$). In addition, for each word in the related concept the cost is already $O(n^3)$. This cost is multiplied by the one derived by the radius. In Chapter 7 we will study the optimal radius value and we will see that in our problem the radius is rather small and does not imply a computational problem.

## 3.5 Discussion

In this chapter we have presented basic concepts in IR and the traditional models and the language models have been introduced. We have reviewed existing methods to reformulate the original user query and the different sources for these reformulations and how the different models are used.

Finally, we have prepared an ontology query model that is integrated into the language models and the ontology. This model allows for different configurations of the probabilities according to different methods or heuristics. Our configuration is prepared to study the effect of changes in the ontology in a method that intends to improve retrieval performance based on specific refinements to the ontology. The algorithm performing ontology refinements is presented in the following chapter.

# Chapter 4

# Ontology Refinement

## 4.1 Introduction

Ontologies developed in the Biomedical Domain collect domain knowledge that has shown to be useful for the domain. But these ontologies are not developed to be used in a specific task for which it is required a more specific content. A refinement of these ontologies may be required to cover the specificities of the task.

Ontology refinement intends to fine tune an ontology and is one of the ontology lifecycle steps. In this step, all the structured sources available to populate the ontology have been used and the refinement has to rely on unstructured sources like text. Techniques that deal with textual data like IE play an essential role.

Considering document collections like Medline, with a high growing rate, techniques that automate the enrichment of existing data sources like databases or ontologies are advantageous. This chapter introduces previous work in ontology refinement based on corpora and introduces our method for ontology refinement. The approaches for ontology refinement supported by unstructured sources are divided in two main groups given the task developed by the ontologists.

## 4.2 Related Work

In this section we consider approaches where textual resources are analyzed. Considering the existing approaches we have categorized the related work into two categories considering the role of the ontologist.

- Semi-automatic approaches in which the refinement algorithms aim to help the ontologist to find the relevant information; reducing the effort of looking for new relevant pieces of information.

- Automatic approaches, which require some heuristic to drive the integration of new knowledge in the ontology.

### 4.2.1 Semi-Automatic Approaches

These approaches provide to the ontologists evidences of knowledge found in the data sources that do not exist in the ontology and that might be interesting to include. In this case, the ontologists does not only rely on their knowledge but are supported by the existing information present in relevant corpora.

We may say that these approaches are recall based since they elicit knowledge from corpora and the main concern is to reduce noise without loosing relevant information. Typically these techniques look for statistically significant terms interesting to the domain against common terms not relevant for ontology learning. A drawback of these techniques is the large number of associations to revise that may not be useful. But these methods provide means to refine more efficiently an existing ontology depending on the quality of the associations.

Some of these methods exploit term co-occurrence to identify new information based on statistical means. Faatz and Steinmetz[47] propose a system where the outcome is a link between existing terms with certain quality that are ready to be analyzed, even though the relation between the terms is not defined. Maedche et al.[103] work on association rules to identify relations between concepts given a set of documents describing hotels . The association study is based on the work of Srikant and Agrawal[150] where the confidence and the support of the associations are used to choose the related candidates. Kohler et al.[93] compare the GO with different ontologies to propose conflicting concepts (e.g. circular definitions) and new synonyms to the ontologists, where the existing relations in the GO are considered.

### 4.2.2 Automatic Approaches

In the previous approach the user plays an important role in the refinement. Automatic methods substitute the ontologist by a process that takes the appropriate integration decision for the terms extracted from corpora that seem interesting for ontology refinement. These automatic methods rely either on heuristics (like some quality measure, Hahn and Schnattinger[66]) or on IE from unstructured sources. The main assumption supporting these approaches is that the documents share the same conceptualization. The evaluation of the techniques is done either by domain experts or by using an existing ontology as gold standard. In addition, the existing techniques that we present in this section present different ways to refine the taxonomy of the ontology but do not propose mechanisms to refine the relations between the concepts.

Navigli and Velardi[108] worked on the adaptation of WordNet to the tourism domain. The idea is to extract new terms from text and then place them in the taxonomy or identifying taxonomic relations between existing concepts. The system uses some heuristics based on term composition and head dependency

of the terms found in the document and on extraction patterns trained on an annotated corpus. The evaluation is done using a gold standard ontology previously built by ontologists helped by semi-automatic methods. In the Biomedical Domain, Lee et al.[97] propose an automated method to refine the Gene Ontology. The idea is to find rules based on GO terms variations for automatic expansion that is validated with the literature.

Agirre et al.[2] refined WordNet to improve their results in Word Sense Disambiguation (WSD). The method consists of adding new terms to the ontology identifying the different senses of the terms and then builds the sense signature using clustering techniques. The performance on WSD is used to evaluate its performance. But the ontology refinement is not driven directly to improve the WSD problem directly.

Some techniques combine different approaches that extract the same information from a corpus [4, 5, 30] increasing the confidence of the extracted information integrated into the ontology. The techniques presented focus mainly on the refinement of the taxonomy and their evaluation is usually performed using a gold standard ontology or measuring the performance in a given task (e.g. WSD) but there is no specific optimization approach to refine it for the given task.

### 4.2.3 Discussion

Ontology refinement approaches rely on facts extracted from text guided by some heuristics, but do not have a relevant task to use it. These refinements are difficult to evaluate, so it is difficult to compare the different approaches. The modifications done to the ontology either are proposed by an ontologist in the semi-automatic approaches or focused in the taxonomy in the automatic approaches. Moreover, the related work presented in the automatic methods develops several approaches to refine a taxonomy but do not provide explicit mechanisms that consider other parts of the ontology. In addition, an ontology has an explicit intention that is not foreseen by these methods. The consequence is that there is no guarantee for the ontology to be useful for a specific task. The cost of having this knowledge may be more expensive than not having it, which has to be maintained and may distract the potential users. In the following section we present our approach for ontology refinement.

## 4.3 Our Ontology Refinement Algorithm

As introduced in the previous chapter, there are some issues that affect the precision and recall of the retrieval that we would like to solve using query formulation and ontology refinement. Recall can be improved, for example, by adding synonyms to a given concept. Precision can be improved by adding knowledge about a known concept, so we can disambiguate the terms related to it in context.

As discussed in Chapter 2, the assumption in the refinement algorithm is that we can measure the relevance of the ontology to the problem by the way the ontology behaves in the task (e.g. IR); so we can compare different ontologies as they perform in the task.

When running a retrieval system we find that some documents are not retrieved and that some documents are retrieved but are irrelevant to the query; these are failures of the retrieval system. In our approach we want to select the appropriate improvements that might repair the ontology. As we see in Figure 4.1, we represent the space of documents D. The documents with a plus sign are relevant documents for our query; with a minus sign are irrelevant documents for our query. We intend to find the missing knowledge to cover the documents not covered and to discard the irrelevant documents enclosed by the retrieval rule R.

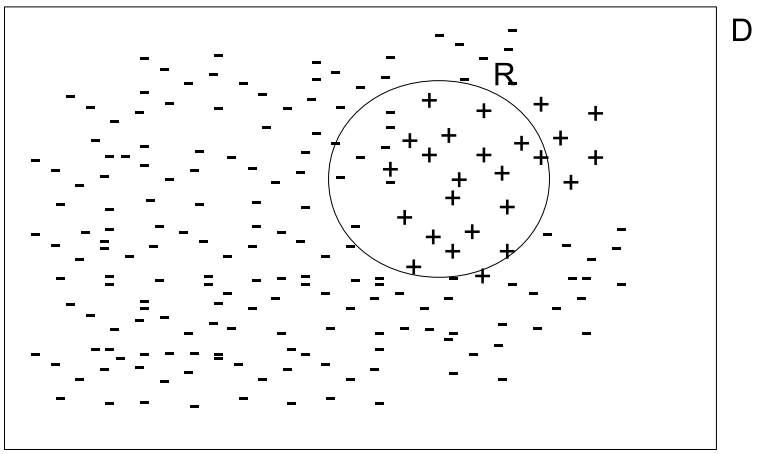


Figure 4.1: Document space (D) and the retrieval result (R)

Ontology refinement defines a search space on which our algorithm has to decide. Without any knowledge/heuristic that prunes all the possible combinations on a huge ontology, we will have a search space of refinements that is too large to be explored. Our approach for ontology refinement considers several heuristics to delimit the search space.

The refinement algorithm has four main components (Figure 4.2). The flaw detector identifies possible flaws of the revised query finding terms in false positives and false negatives. The refinement generator turns the flaws into ontology refinements linking facts extracted from the corpus using IE. The credit assignment process filters out the proposals in conflict with the ontology being refined and gives credit to the remaining ontology refinements. Finally, the ontology refinement space is traversed to find the ontology refinements that may improve the performance of the system and selects the nodes to produce the following node expansion in the search space.

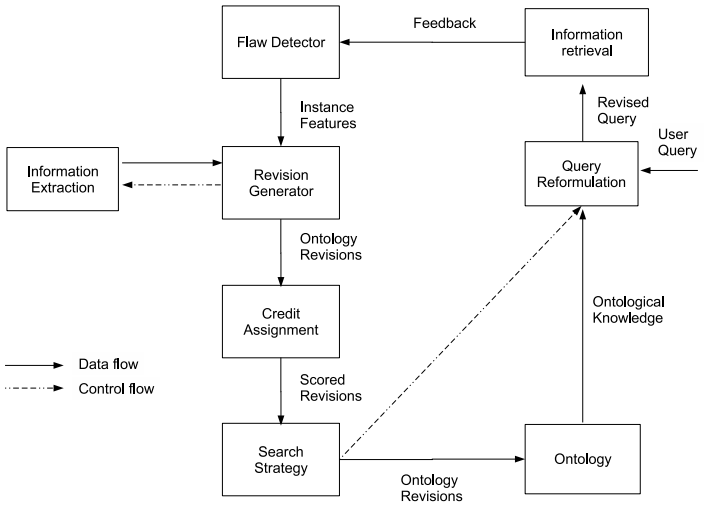


Figure 4.2: Refinement algorithm

### 4.3.1 Flaw detector

This process analyses the false positives and false negatives to identify which terms are interesting to be considered in the query formulation process. Our assumption is that there is a set of terms that if added to the query formulation through the ontology will have an impact on precision and recall. This process does not solve yet the refinement of the ontology since the information carried by these terms need to be interpreted and included in the ontology. Our system knows what the expected output is and will try to guess which terms in the documents may be representative of the missing documents or the irrelevant ones.

The flaw detector has two subcomponents, the feature extractor, which decomposes documents into a set of features that the IR approach can process and the feature selection which analyzes the documents as categorized in a contingency table to find more relevant features to be considered. The flaw detector pipeline appears in Figure 4.3.

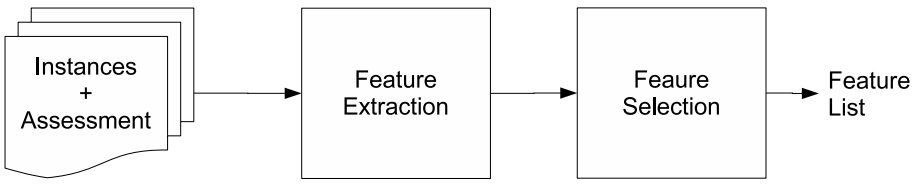


Figure 4.3: Flaws detector

**Feature extraction** The feature extractor turns the documents into a set of features that can be used in the IR model. Several possible representations appear according to the retrieval approach. As we will see in Chapter 5, we extract the terms doing a syntactic analysis based on shallow parsing provided by *ltchunk* software developed by Andrei Mikheev of the University of Edinburgh. Removal of stop words is performed to reduce noise and POS (Part-of-Speech) filtering is used to remove terms like determinants.

**Feature selection** The number of features extracted from the documents tends to be quite large and not all of these features are relevant. The documents are transformed from a textual representation into a representation similar to the bag of words. Different statistics can be estimated to select the set of the most relevant terms as information gain or the f-measure. In order to reduce the possibility of noise in the data set, the extracted terms for which the document frequency is under a given threshold or appear in a stopword list are not considered, so we reduce the size of the search space by discarding common terms and hopefully the risk of overfitting due to some rare terms. Several criteria may be applied to filter the terms which will be provided to the following step. The heuristic measure estimated for each one of the terms can be used to either select the first n terms ranked by the heuristic measure or to select the set of terms above a given threshold in the heuristic measure. A study of the different options is presented in the results section.

### 4.3.2 Ontology Refinement Generator

The terms selected in the previous section give hints of the flaws in the query formulation approach and may identify flaws in the ontology. Before proposing changes in the ontology we have to relate the terms into ontological knowledge. In the ontology refinement process we will find the possible ways the term can be integrated in the ontology. In order to explicitly state the changes that need to be done, a set of ontology refinement operations have been enumerated. We have exploited the different resources and the only resources available are document collections, but they are unstructured sources. IE is used to extract facts from the literature. In this section we explain the procedure that we followed to turn the set of extracted terms into candidate ontology refinements.

The features extracted in the previous process tell us where to look for ontology refinement. In this section we propose a methodology to identify, given a set of terms, the possible refinements supported by IE; linking the extracted facts into modifications to the ontology.

**Operations on the ontology** We define the possible atomic operations that can be applied to the ontology with a set of operators defined in Table 4.1. Even though the problem can be defined as a refinement where we can add and remove knowledge from the ontology; any modification can be seen as a composition of adding and deleting knowledge from the ontology and the lexicon. In our

work we consider only the refinement of knowledge from the ontology and do not consider any pruning of the already existing knowledge. This means that from the operations presented in this section we consider the ones that solely add new knowledge to the ontology. The only exception that we will find in the results chapter is related to the cleaning of the lexicon linked to the ontology.

| Instruction | Description |
|---|---|
| AddL($l$)/DelL($l$) | adds/deletes the term $l$ in the lexicon $L$ |
| AddF($l$,$c$)/DelF($l$,$c$) | links/unlinks the term $l$ from $L$ with the concept $c$ in $C$ |
| AddC($c$)/DelC($c$) | creates/deletes the concept $c$ in $C$ |
| AddHc($cp$, $cc$)/DelHc($cp$, $cc$) | adds/deletes a hierarchical relation in $Hc$, where the concept $cp$ is a more general concept than $cc$ |
| AddR($r$, $c1$,$c2$)/DelR($r$,$c1$,$c2$) | adds/deletes the relation $r$ between $c1$ and $c2$ |
| AddG($l$,$r$)/DelG($l$,$r$) | adds/deletes the link between the term $l$ and the relation $r$ |

Table 4.1: List of operators used to modify the ontology

For instance, if we want to add the term *DNA repair* to the ontology and link it to the gene *MMS2* we will require the following operations:

| Instruction | Description |
|---|---|
| AddL("DNA repair") | adds the term in the lexicon and returns the identifier $l1$ |
| AddC($c1$) | adds the concept $c1$ in $C$ |
| AddF($l1$,$c1$) | links $l1$ to the concept $c1$ |
| AddR($r1$,$c1$,$c2$) | links between the concept $c1$ and the concept MMS2 $c2$ with the relation $r1$ |

Table 4.2: Operator example for DNA repair and MMS2

A decision process is used that takes a term and provides, based on IE, a list of the defined operations.

**Decision process** A decision process is used to transform the IE filled templates into a set of operations to be performed on the ontology. This idea is similar to Sintek et al.[144]; where the different steps for the integration of terms is similar to ours but we rely on a specific decision process while the path in the other methodologies has to be specified. This decision process would do similar tasks to the ones that are already performed by the ontology engineer.

The input to the decision process is a term provided by the flaw detector and the output is a sequence of operations that will integrate this term or relations to related concepts in the ontology. Figure 4.4 shows the decision process. In

this decision process, the first condition to verify is if the term is missing in the ontology lexicon (L). This can be implemented using a look-up function. Term recognition techniques have already been studied in the literature as can be found in the work by Jacquemin[77] and more specifically in the Biomedical Domain with MetaMap[8] or Whatizit[123] for named entity recognition and resolution.

If the term is not in the ontology lexicon, the term is included in the lexicon with the AddL operation. Then we have to consider if the term is a synonym of an existing concept or if the term belongs to a new concept, not existing in the ontology. If the term within its context can be assigned to an already existing concept, the link between the term and the concept is added with AddF. If we find that a new concept is needed, this new concept is created, placed in the taxonomy of concepts by finding the most appropriate hypernym in the ontology and finally, the term is linked to this new concept. Then, we try to find the relation of this concept with the existing concepts in the ontology. We have identified several tasks for IE in the decision process. These tasks are: synonym relation, taxonomic relations and no-taxonomic relations. Existing IE techniques that can be used for these tasks are defined in Chapter 5.

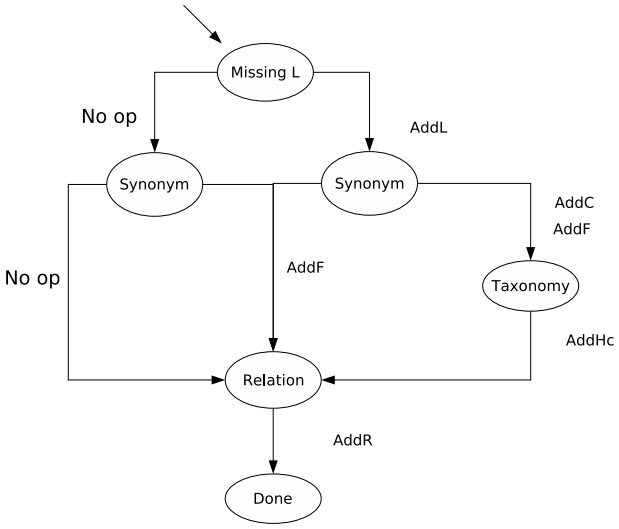


Figure 4.4: Decision process. *No op* means that no operation is done.

In Table 4.2, we find an example where we consider creating the concept *DNA repair* which is labeled with a new term added to the lexicon that we link to the protein *MMS2*.

### 4.3.3 Ontology Refinement and Credit Assignment

The refinements suggested by the ontology refinement generator offer the possibility to solve the flaws of the ontology to fulfill the retrieval task. On the other hand, these refinements may still not be relevant for the retrieval task or the proposal may conflict with the ontology being refined. A filtering of conflicting proposals and an assignment of credit for the refinements is needed.

The refinement proposals may conflict with the ontology because of undesired loops or undesired entailments. The loops in the ontology have a negative impact in the ontology query model presented in chapter 3. A filter is used to avoid introducing loops to the ontology. The rejected proposals are not any longer considered in the following processes.

We evaluate the different ontology refinements and choose the most appropriate for the task, this differs from the work of Brank et al. [19] since we want to find an ontology with the best performance instead of knowing if the ontology is appropriate for the task. The utility is obtained by calculating the evaluation function; in this case it is the performance of the query reformulation rule over the queries.

### 4.3.4 Search strategy

A search strategy can be applied in order to find a refined ontology. [1] Blind methods like breath-first or deep-first will evaluate too may cases and, as in the case of decision trees, are impractical. Iterative heuristic methods are more appropriate because they are guided by some heuristic. In the experiments we will start with the Hill-climbing search strategy (c.f. Algorithm 4.1) which takes the point with the maximum value as a heuristic measure and stops where there is no higher point. This algorithm considers the different steps of the refinement algorithm to generate candidates and uses the search strategy to select the refinement with the largest improvement.

This search strategy is simple and fast, but there is risk of local optima. For a given ontology the algorithm will look for the refinement with the higher value in the evaluation function and this refinement will become the new ontology, so the ontology refinement generator will generate new proposals. Other search strategies can be used like simulated annealing, tabu search or genetic programming to overcome the local optima problem. Specific optimizations of the search space are presented in the experimental sections.

[1] Ontologies may be large. Different versions of an ontology cannot be stored due to space reasons. In order to solve this problem and have the search space with the different ontologies, instead of having different versions, only the operators needed to have the revised ontology are stored and used it is needed for evaluation. The operators for each refinement are applied, evaluated and then removed from the original ontology.

**Algorithm 4.1** Ontology refinement step using Hill-climbing

1: Run the document retrieval based on the ontology version i on a set of queries.
2: Evaluate IR performance.
3: Do credit assignment to identify which parts of the ontology need to be revised in order to improve the performance. Select a set of the proposed refinements.
4: Execute the refinements identified in step 3 and install the revised version (i+1) of your ontology in preparation for the next cycle.

## 4.4 Discussion

We have presented an algorithm that benefits from the feedback given to an IR system to propose modifications that may improve the behavior of the system in unseen events. The evaluation of the ontology based on IR performance has been used to decide on the possible directions to optimize the ontology. In addition we have presented an approach in which IE can be integrated. The cost of the algorithm is defined by the sum of the cost of the different components. Due to the size of the search space, the cost is quite relevant.

# Chapter 5

# Information Extraction

## 5.1 Introduction

In Chapter 4 we have introduced IE as a source to suggest ontology refinements. IE gleans facts from unstructured sources (e.g. documents). IE is different to IR since the first one extracts facts and the latter recovers documents from textual sources.

In the following example sentence, we are interested on an extraction need expressed as a template. Its slots are filled by the IE system. Several analyses are performed (cf. Figure 5.3) that produce a structured output expressed by the template.

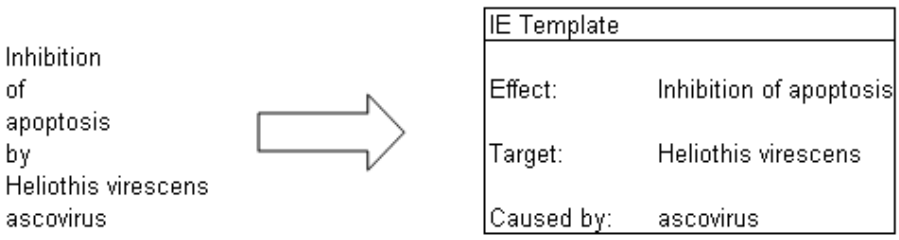


Figure 5.1: Information Extraction example

In text mining systems, IR and IE usually interact (e.g. Figure 5.2). IR is used to retrieve relevant documents or sentences to be processed by IE, while IE may feed IR to produce better indexes. In our case IE is used in the refinement of our ontology to improve the query model.

The development of an IE system follows either the knowledge engineering (KE) or the machine learning (ML) approach; even though mixed approaches co-exist[135]. In the first one, the IE patterns are developed by a knowledge engineer, usually a domain expert. The quality of the produced rules depends on

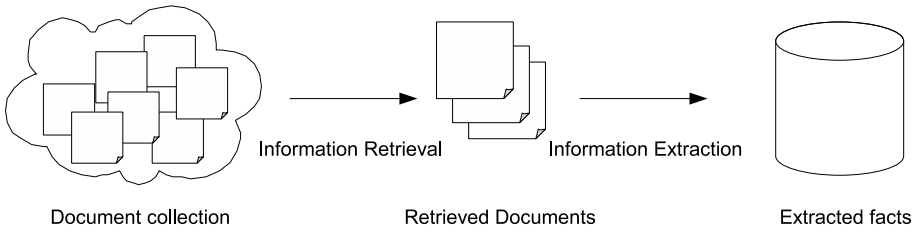


Figure 5.2: Information Retrieval and Information Extraction interaction

the ability of the engineer to master the features of the system. Machine learning approaches use available training data to produce extraction patterns. The representation of these data in order to produce the patterns and the availability of the training data are the main challenges of this approach. Compared to the knowledge engineering approach we only need training data that may be cheaper to collect than the generation of rules by an engineer.

The KE approach needs to rebuild the set of extraction patterns if the interest moves to a different domain. On the other hand, the domain knowledge of the knowledge engineer may produce extraction patterns that provide better performance since they are built based on experience and are easier to maintain for the same task since the rules are usually easier to understand than the rules produced by ML approaches.

Several tools exist in the field of natural language processing and IE. One of the main problems is that their interaction is not easy. Several systems are available that integrate different components for IE. A popular system is GATE [1] that integrates many natural language processing and IE tools. The UIMA framework [2] allows the integration of different systems. There is an initiative that allows the integration of different tools enhancing the original document based on XML called IeXML[125]. In this initiative, tools only need to comply with the standard and work in pipeline mode, being easy to interchange the components.

IE approaches are difficult to compare since they deal with different extraction needs that are reflected in the available data sets[22]; some of them are publicly available. In the Biomedical Domain, several data sets are freely available for different IE tasks like the Biocreative I and II, the GENIA corpus, the BioInfer dataset, AIMed, Prodiser and weakly-annotated like Craven's dataset. These datasets are far from covering the domain and the focus is mainly devoted to the identification of protein/gene names (PGN), the functional annotation of proteins and the interaction between proteins (PPI).

Even though the requirements for extraction depend on the application, IE systems usually have the same components[35] that can be combined in pipeline as shown in Figure 5.3. This pipeline approach modularizes the IE systems allowing the interchange of several components. As an engineering artifact, a

[1] http://gate.ac.uk/
[2] http://www.research.ibm.com/UIMA

common representation will guarantee the inter-operability of the components and several initiatives exist. Typical components in an IE system are:

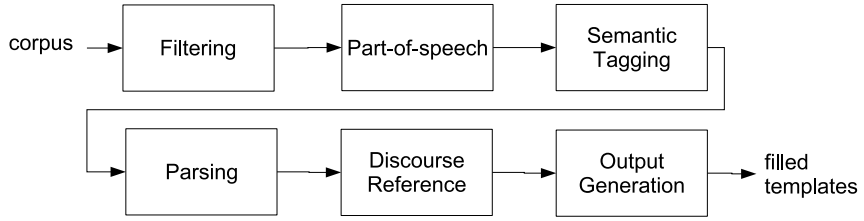


Figure 5.3: Information Extraction components

**Filtering** Performed at the text level, the idea is to select the pieces of information relevant for further processing. Traditional IR systems or text categorizers may be used to do this selection.

**Part-of-speech** It assigns the part-of-speech (POS) to the words in the filtered text. This allows, in later stages, to perform decisions using this annotation like word sense disambiguation or parsing. Table 5.1 shows the example sentence tokenized (tag $w$). For each token, we find the POS in the Penn Treebank annotation (attribute $c$).

| |
|---|
| <w c='NN'>Inhibition</w> |
| <w c='IN'>of</w> |
| <w c='NN'>apoptosis</w> |
| <w c='IN'>by</w> |
| <w c='NP'>Heliothis</w> |
| <w c='NP'>virescens</w> |
| <w c='NN'>ascovirus</w> |

Table 5.1: Part-of-speech example

**Semantic tagging** Identification of major phrasal units that may be done using for instance shallow parsing (driven by the POS) as we find in Table 5.2 or based on named entity recognition techniques as we find in Table 5.3.

**Parsing** The idea is to identify the relation between the different units provided by the previous analysis at the sentence level. Several parsers are available trained usually on standard corpora like the Penntree bank. Several efforts exist in the Biomedical Domain like the Enju parser which uses the GENIA corpus. In Figure 5.4 we find the example sentence processed by Enju.

| |
|---|
| <c c='NP'>Inhibition</c><br>of<br><c c='NP'>apoptosis</c><br>by<br><c c='NP'>Heliothis virescens</c><br>ascovirus. |

Table 5.2: Shallow parser example

| |
|---|
| Inhibition of<br><e id='GO:0006915' onto='biological_process'>apoptosis</e><br>by<br><e id='species:7102'>Heliothis virescens</e><br><e id='species:43680'>ascovirus</e>. |

Table 5.3: Entity annotation example

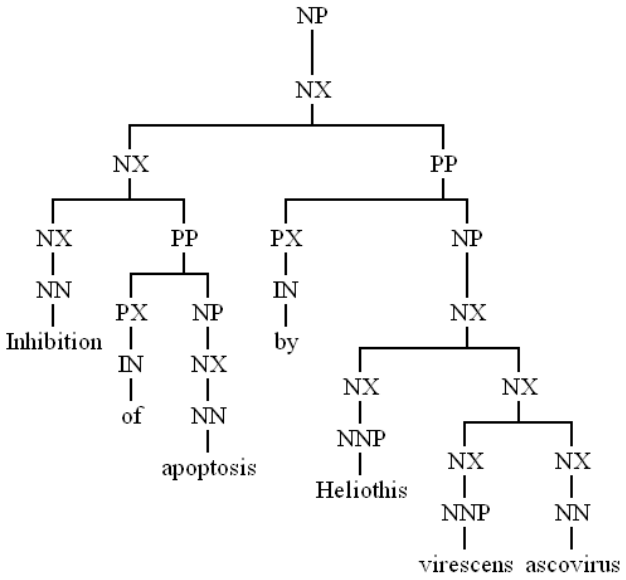


Figure 5.4: Enju parse example

**Discourse Reference** Some facts are expressed across sentences. In this component the output of the different parsed sentences are related, recognizing and unifying referring expressions. Techniques like coreference resolution are used to identify phrases denoting the same entity expressed across sentences.

**Output Generation** The IE task consists of extracting facts from text. The task can be specified as filling a template where the different slots are the related entities. The outcome of the previous components is prepared to fill the specified template.

In the following section we highlight IE tasks relevant for ontology refinement with different approaches identified in the literature. For each of these tasks we will present the current approaches and we will introduce the implementation used in our work.

## 5.2 Information Extraction Evaluation

IE approaches are evaluated using the measures presented in this section. In the different tasks we will usually assign to pieces of text a label (annotation), like the part-of-speech or the semantic type of the entity, or will fill slots with information extracted from text. We will compare this annotation to a reference corpus. The annotations that match the reference corpus will be considered as *true positives*, while the annotations that are not present in the reference corpus will be considered as *false positives*.

Precision measures the ratio of true hits compared to the false hits in the annotated set.

$$Precision = \frac{true\_positives}{true\_positives + false\_positives} \tag{5.1}$$

Recall measures the ratio of true annotations among the known labels or slot information.

$$Recall = \frac{true\_positives}{true\_positives + false\_negatives} \tag{5.2}$$

The F-measure is a ratio that combines precision and recall.

$$F\alpha = \frac{(1+\alpha)Precision * Recall}{(\alpha * Precision + Recall)} \tag{5.3}$$

Typically $\alpha$ is set to 1; meaning that we do not prefer precision over recall:

$$F1 = \frac{2 * Precision * Recall}{(Precision + Recall)} \tag{5.4}$$

## 5.3 Information Extraction and Ontology Refinement

In the previous chapter we introduced a flaw detector and a decision process that produces ontology refinement candidates. The flaw detector requires the identification of new and already existing terms to be used within the decision process. The decision process relies on information extracted from text for which IE is used. The following section enumerates these IE tasks and presents the state of the art approaches. The specific required tasks are:

- to find out if a given term is a synonym of an existing concept involves the identification of the concept denoted by the term (NER) and the candidate synonym concepts. If no specific concept is found, a new concept is created.
- to find the parent of a given concept (taxonomic relations).
- if there is a relation between a concept and any other concept in the ontology (non-taxonomic relations).

These tasks rely on identifying pieces of information relevant for a given context defined by the term we are processing. Medline contains several million entries and it is very expensive to run the approaches on the whole of Medline. A selection of citations is done by IR. In addition, the sentences belonging to the retrieved documents will be ranked by relevance whenever an appropriate model can be produced. This model will depend on the extraction patterns that we intend to use. If an appropriate approach for sentence ranking is not found, a Boolean expression based on the terms in the pattern is used to retrieve only on potential relevant sentences and speed up the extraction system.

As mentioned above, one of the technical problems to reuse the available tools is the lack of a common representation that allows an easy integration of the components. We use a proposed XML standard format named IeXML that can be embedded into the Medline XML[125].

## 5.4 Term Extraction

We are interested in identifying known concepts in text and in the identification of new relevant terms not existing in the lexicon of the ontology. In contrast to named entity resolution, where the lexicon linked to the ontology may be used as the source of terms, we require to extract possible valid terms from the corpus that are relevant to the domain.

During the different stages of the lifecycle, the lexicon will provide the terminology for existing concepts. If there is no entry in the lexicon, the current process may suggest the creation of this new entry.

Terminology management concerns the creation, storage, maintenance, updating and curation of a terminological resource. We will show two complementary approaches of methodologies used to create these resources proposed

or implemented for the Biomedical Domain. The first one is an automatic approach that relies on a document collection and the second relies on existing resources.

A proposal for automatic term management (ATM) can be found in Ananiadou and Nenadic [7]. This approach is composed of three modules. The first module is automatic term recognition, identifies lexical structures that can be mapped to domain concepts. The second module implies term structuring that identifies relevant relations or associations between terms mainly using classification and clustering. The last module consists of an intelligent term manager that in addition of storing the terms accordingly it may provide links and definitions to existing resources. Existing resources can be reused to train the classifiers or to use them in dictionary approaches to term recognition.

In addition to this approach, we can use approaches that collect existing structures from available terminological resources. The development of the UMLS, as introduced above, is a merging of several resources. Several issues appear when merging the different structures, being similar to ontology alignment problems. This approach will require ATM solutions to extend and maintain existing resources once the available ones have been used.

The selection of terms for this entry requires the use of appropriate terms. These terms may be provided by a community effort where several domain experts study the appropriate set of terms and/or using natural language processing and text mining[149] to extract terms from the literature[53]. Tools are available to find terms in context to verify their use like keyword in context concordancers[92].

Term extraction can be done using a set of patterns based on the part-of-speech linked to each word (like Adjective Noun, ANN, ...) and using an expression that allows to select valid terms. Then statistics over large corpora are estimated, an example of this technique is provided by the C/NC-value[53] in the Biomedical Domain.

Another option is to use a parser to detect dependencies between words, such as a shallow parser or chunker to identify noun phrases denoting terms versus statistical analysis of Medline for word co-occurrence. A problem may appear in the case of prepositional attachments.

As a given term may accept different variants in text (*cancer colon* vs. *cancer of the colon*). Morpho-syntactic normalization [77] might be required to provide better statistics on terms.

Terms are extracted from the document set relevant to our retrieval task and selected and ranked according to their distribution. These terms may denote lexical strings to be added to the ontology. We have decided to use an already existing system to perform the analysis of the documents and extract the candidate terms according to the analysis of the dependencies of the terms named LTChunk. This application performs a POS analysis of the documents, then verb and noun phrases are identified. An example is shown in the following figure:

```
<text>
```

```
  <s><c c="NP"> Sulindac </c> <c c="VP"> sulfone is most </c>
  effective in modulating <c c="NP"> beta-catenin-mediated
  transcription </c> in <c c="NP"> cells </c> with <c c="NP">
  mutant APC </c>.</s>
</text>
```

The extracted terms are normalized to improve the statistics estimated on the document set. In addition to the term extraction approaches proposed in this section, a NER system performs an annotation of existing entities. This allows us to detect terms that are recognized as existing entities.

## 5.5 Synonym Identification

We consider that two terms are synonyms when these terms are equivalent in meaning and where one side can be substituted for the other in a specific context without loss of meaning. The different techniques to identify synonyms are either, based on the inner structure of the term or based on the context of the term (syntactic or statistical). The results of these systems show that there is still a lot to investigate on this subject, showing the difficulty of the task.

Methods based on the inner structure use mainly term matching techniques. Hole et al.[74] link terms if they share any word like *cerebrospinal fluid* leads to *cerebrospinal fluid protein assay*. Techniques using variation of terms exist like trigram-matching algorithms to identify similar multi-word phrases. Terms are treated as documents made up of character trigrams. A vector space is built and similarity is computed by Wilbur and Kim[163]. These techniques try to maximize recall while having a poor precision performance. It is up to the user to find out which of the related terms are really synonyms.

Based on the context of the term we find different techniques. Dagan et al. [40], Li and Abe [98], Lin [99] relate two terms calculating the similarity measure of the contexts. Yu et al. [169] make a study based on these techniques to find synonyms for genes and proteins without great success. Based on machine readable dictionaries (MRD), Resnik[126] uses WordNet to evaluate semantic similarity of any two concepts based on their distance to other concepts that subsume them in the taxonomy.

Pearson [114] found several patterns like *known as, called as* useful for synonym identification. This method is simple but does not make a clear distinction between synonyms and hyperonyms. Yu et al.[169] introduces new semi-supervised methods on the basis of large corpora and few examples, using bootstrapping based on the SNOWBALL system to learn extraction patterns indicating synonymous relations.

Terms extracted that cannot be identified as existing entities may denote either new strings for existing concepts or new concepts. As we have seen before, context based approaches have not shown an interesting performance compared to manual systems based on rules. We use several methods to identity synonyms. The first one is based on patterns to identify synonyms denoted explicitly as we can see in Table 5.4.

| Rule |
|---|
| NP know as NP |
| NP is know as NP |
| NP is also know as NP |
| NP called NP |
| NP also called NP |
| NP called as NP |
| NP also called as NP |

Table 5.4: Synonym rules

Acronyms are another source of synonyms. Acronym databases exist which can help to identify the possible long forms. We are interested in the extraction of the specific long form used in context, so we require an analysis of the specific context and we will rely on the probability of finding the specific long form in our specific context. So the frequency that a given acronym has been linked with the long form in the document set is required.

The long and short form candidates in the documents are selected using specific regular expressions that detect the presence of the acronym close to its expanded form and basically rely on the search of a noun phrase and an expression in parenthesis. Then the longest of these two expression will be assigned the label long form and the other one the label short form. The method defined by Schwartz and Hearst[141] is used to estimate a score indicating a kind of probability of the acronym being generated by the long form based on fuzzy matching. This method is simple and effective and no training data is required. In[141] there is a sample implementation in Java.

Different patterns that have a higher probability of happening in the document set are considered to be used in the refinement process.

## 5.6 Named Entity Recognition

### 5.6.1 Introduction

Named entity recognition and resolution of the entities to existing resources like databases or ontologies are relevant to identify already known concepts with potential interest. These methods have to consider the problems that appear in the identification of entities in text: synonymy, ambiguity and variability. In the Biomedical Domain, there are few available systems that cover the available semantic types. Among these systems we find MetaMap[8] provided by the National Library of Medicine (NLM) that normalizes terms into UMLS concepts and Whatizit [123] provided by the European Bioinformatics Institute (EBI). Our IE system relies on the Whatizit system and mainly relies on dictionary look-up techniques with disambiguation of the annotated entities. The evaluation of these systems is subject to the availability of annotated corpora. There are different methods that allow us to perform these two tasks that are presented

in the following sections.

**Dictionary look-up** This technique matches the terms in the lexicon to the terms in text. Pure look-up techniques are not robust against term variability not foreseen during the development of the lexicon. Different approaches are used from exact match to more refined methods using natural language processing[77], variant generation and statistical means[84]. One drawback is that the system does not rely on the context to discard false positives and it is usually combined with other techniques like machine learning[115].

The processing of the dictionary is needed to clean up noisy terms or redundant terms that will not have a representation in text. In the Biomedical Domain, there are large terminological resources like the UMLS Methathesaurus or databases, like Uniprot, that carry many terms related to proteins and genes.

**Rule based** Follows the knowledge engineering approach, the knowledge engineer prepares the set of rules required for the identification of entities. This approach is capable of dealing with a larger set of variants than the dictionary approach since it is possible to develop rules that consider the context of the entities. The first approach applied for protein named entity recognition is presented by Fukuda et al.[55]. First the method identifies terms of interest like *protein* or *receptor* and then hand-crafted rules are used to extend the names to capture the missing terms.

**Machine learning** Relies on the use of machine learning algorithms. Even though Hidden Markov Models (HMM) were the most popular algorithm used, in recent years there are techniques like conditional random fields that have provided better performance in the recognition of genes. The problem with these techniques is the lack of training data. Several approaches exist to collect it, either generating noisy data or combining different machine learning algorithms in adaptive learning approaches where a small set of training data is used to train different algorithms and the users only have to correct the annotations in which the algorithms disagree the most.

**Hybrid approaches** Sometimes there is the need of combining different systems since they share different views on the same data. For instance, we have developed a hybrid system[115] which combines a dictionary approach and a machine learning approach.

### 5.6.2 UMLS processing and Disease Annotation

We have contributed to the Whatizit system with annotation systems based on the UMLS Metathesaurus; mainly on the evaluation of disease annotation[84]. The corpus provided by Mark Craven, that consists of OMIM references, is used to extract entities from text. We have evaluated the resolution of disease name

entities and resolution based on the UMLS Metathesaurus which allow us to compare the MetaMap with other techniques.

**Processing of the UMLS Metathesaurus** Our UMLS version used in this project is 2006AD. The preparation of the lexicon requires a number of steps. The first one is to select the sources that our local installation of UMLS will contain. Then the lexicon related to the diseases has to be extracted and filtered since some terms may be too ambiguous or not representative of the disease in text, thus redundant. Different sources contribute to UMLS. Some of these sources contain terms that are less relevant for term recognition such as numbers or single letters. We filtered the UMLS Metathesaurus Lexicon according to the steps proposed by Aronson [3]. In particular, we selected only the subpart of UMLS that covers the English language. Other entries have been filtered out since they are tagged as being obsolete content (flag 'O'), non-obsolete content but marked suppressible by an editor (flag 'E') or non-obsolete content but deemed suppressible during inversion (flag 'Y'). Concept terms with ten or more words have been deleted, since we believe that they are not used in scientific literature and thus their information content is very low for this study. Finally, we collected general terms in a stop word list that includes terms like *disease* or *syndrome* that do not to provide new information. We also removed terms contained in parenthesis and attached to the lexical item, e.g., Neoplasm of abdomen (disorder). With regards to ambiguity, the Metathesaurus has also been processed to solve some ambiguous cases, that is, strings that have two or more assigned CUIs (Concept Unique Identifier). We distinguish three other types of term ambiguity. The first one is discussed by Aronson and Shooshan[4]. They present a set of ambiguous concept names with degree 40 to degree 6 (the degree is the number of different CUIs associated to the same term string). We have followed their work in order to detect and delete the suppressible ambiguous cases. The second case of ambiguity is the one involving concepts with different semantic types and not covered in the previous method. Cases like brain as a synonym of brain disease are representatives of this type of ambiguity. Priority was given to semantic types relevant to this project. The third case of ambiguity involves concepts from the same semantic type. This ambiguity is not solved, and the term will have associated a set of CUIs, that is, if the string is detected in text it will be tagged with all related CUIs; as in prostate cancer (C0600139, C0376358). Finally, the terminology has been filtered semantically to select the sources for the disease lexicon. This is done based on the UMLS semantic types. We have selected terms belonging to the following semantic types: *Disease or Syndrome, Neoplastic Process, Congenital Abnormality, Mental or Behavioral Dysfunction, Experimental Model of Disease and Acquired Abnormality.* After all the processing was done, we had selected a set of 275.000 terms that can be mapped to 85.000 concepts from the different semantic types selected. This provides an average of three terms per concept.

[3] http://skr.nlm.nih.gov/papers/references/filtering06.pdf
[4] http://skr.nlm.nih.gov/papers/references/ambiguity06.pdf

We have compared three different existing methods that perform annotation based on the UMLS Metathesaurus:

**Dictionary look-up** This method matches the terms as they appear in the terminology so it is not robust against term variability that has not been foreseen during the creation of the lexicon. In addition, precision may be affected by ambiguous terms or nested terms and some techniques are proposed to solve these issues.

**Gaudan's statistical approach[58]** Information-theory based approach on which some properties of the lexicon, like the frequency of the words, allow us to calculate: the specificity of the term; the evidence, the number of matched words contrasted with the number of terms in the zone and the proximity of the lexicon terms and the occurrence in the text are analyzed on a specific zone. The zone can cover a noun phrase, a sentence or a whole abstract. The selection of the zone is crucial. For example, if you decide to work at the noun phrase level it may happen that your term is broader than the noun phrase limits. In contrast to the other two methods there is more flexibility on the matching.

**MetaMap** MetaMap is a state of the art approach for semantic enrichment of the literature. Metamap splits the text given as input into sentences, and the set of sentences into phrases (noun and verb phrases). For each phrase Metamap identifies possible matches and for each match the set of possible candidates to match with a concept name from their filtered UMLS Methatesaurus[8], associating a score to each of them. Metamap uses exact match and partial and complex ones; syntactic variations of terms (e.g.: arthritis and tumours). This explains why Metamap identifies several possible matches in each phrase and several candidates for each one. However, due to its flexibility, in some cases the text to be matched is not precise with respect to the UMLS concept.

The corpus for disease resolution has been produced from a subselection of Craven's corpus where initially the boundaries of the diseases were annotated. We could not reuse the annotations since the diseases covered in the corpus are based on the list of OMIM diseases. The corpus has been annotated based on the different methods and then verified by two domain experts where the conflicts have been solved[84].

| | B | Dif | A | Dis | FP | TP | R | P | F-m |
|---|---|---|---|---|---|---|---|---|---|
| D look-up | 586 | 269 | 542 | 217 | 143 | 399 | 68.09 | 73.62 | **70.74** |
| Statistical | 586 | 269 | 760 | 299 | 313 | 447 | **76.28** | 58.82 | 66.42 |
| MetaMap | 586 | 269 | 413 | 182 | 95 | 318 | 54.27 | **77.00** | 63.66 |

Table 5.5: Disease resolution results (Caption: B=Benchmark, Dif=unique diseases, A=Annotated, Dis.=Diseases, FP=false positives, TP=true positives, R=Recall, P=Precision, F-m.=F-Measure)

As we can see in Table 5.5, the dictionary look-up offers competitive results compared to existing methods. This behavior is different compared to other semantic types like the *proteins* and *genes* where a disambiguation component is compulsory to obtain interesting results. As expected, the statistical approach has higher recall with much lower precision. MetaMap is the method with the highest precision but with lowest recall due to a more complex processing and because it works with all the UMLS which allows it to discard entities belonging to other semantic types.

## 5.7 Taxonomic Relation Extraction

Similarly to synonym detection, the different methodologies can be classified depending upon whether they use the inner structure of the term or the context in which the term appears.

Term composition, like head-modifier, can give hints about taxonomy relation, for instance *colon cancer* is more specific than *cancer*. Navigli et al. [108] and Missikoff et al. [106] extend WordNet with domain specific information about the tourism domain based on term composition of multiword terms extracted from domain specific documents.

The context of the terms can be used to identify similar terms. The terms within a fixed length window surrounding a term can be used to prepare a *word sense* model. Hearst and Pedersen [71] collected from the context of the terms some words and built a vector space that was reduced with the principal component analysis (PCA). The result for the task of identifying hierarchical relations is 58% in precision. A variation from Widdows [162] added the Part-of-Speech (POS) to better discriminate the words defining the context.

Some models are based on extraction patterns but the lack of training is one of the biggest challenges. To overcome the lack of training data several methods, like bootstrapping, have been used. Riloff et al. [128, 127] and Roark et al.[130] build dictionaries extracting terms and assigning them to general categories, their method does not scale to more refined categories. Hearst [67, 68] uses a set of very reliable initial extraction patterns that are combined with bootstrapping to find more extraction patterns and hyponym/hypernym relations without a dictionary. Their system obtained very low recall but high precision.

Methods relying on the context have an interesting recall level but suffer from low precision and it is more difficult to differentiate between synonymy/part-of/hierarchical relations. This is interesting if a user revises the produced tree in the case of using a hierarchical clustering approach[16]. On the other hand, our main concern is precision so we would like to consider term composition and some specific patterns to identify the hierarchical relations.

Term composition, like head-modifier, is relevant in several semantic types like diseases where composition seems to be relevant to find the ancestors in the hierarchy. Normalization techniques applied in the previous section contribute to improve the coverage of this technique.

$$coloncancer - is - a - > cancer$$

The Hearst patterns have high precision in contrast to the statistical methods presented above.

| Rule |
|---|
| NP0 such as NP1, NP2 ..., (and \| or) NPn |
| such NP as NP,* (or \|and) NP |
| NP ,NP* , or other NP |
| NP ,NP* , and other NP |
| NP , including NP,* or \| and NP |
| NP , especially NP,* or \| and NP |

Table 5.6: Hearst patterns

## 5.8 Non-Taxonomic Relation Extraction

### 5.8.1 Introduction

We are interested in techniques that help us to find if two entities are related. Despite the most general part-of relation, these relations are domain specific. In the Biomedical Domain, the interest is focused on gene-disease and protein-protein interaction (PPI). Different number of systems have appeared recently with the intention of filling the existing PPI databases: ALFA[156], EMPathIE [76], BioAnnotator [152], PIES [165], BioRAT[41], GeneWays [5],MedScan [6].

A simple method is based on association rules[150] used by Maedche[102]. The drawback is that the label of the relation is not given, this being the task of the ontology engineer. In addition, this methodology can produce many false positives or few true positives.

Other methods based on extraction patterns in the Biomedical Domain are still based on the knowledge engineering approach, like the Bioie [91], RLIMS-P[75], Blaschke[15], Rindflesch[129].

Machine learning examples of supervised systems are used where training data is available (e.g. the works of Cardie [27], Freitag [54], Califf [23, 24] and Donaldson[45]). As the training data is expensive to collect, different methodologies exist depending on how the instances for training are selected. Craven et al. [36, 37] worked on finding the relation between a protein and a subcellular location. The methodology to collect positive examples consisted of selecting sentences where already known relations from a database can be found, a kind of "weak-labeling" as defined by Craven. Active learning has been used [51, 88, 153], semi-supervised like Krogel et al. [95, 94], or based on the refinement of patterns, Ciravegna [31]. Agichten and Gravano [1] develop a bootstrap

[5]http://geneways.genomecenter.columbia.edu/
[6]http://www.ariadnegenomics.com/products/medscan

approach but the way they built their patterns has two main drawbacks [134]. The first one is that there is an assumption that there is only a possible relation between two entities mainly because they consider simple relations and the second problem is that the patterns that the system can build have a predefined structure which does not always happen as in real documents. In the following section we present our work on co-occurrences complemented with a filtering system that discards sentences that are not relevant for the relations of interest.

### 5.8.2 Co-occurrence analysis

As we have seen, different methodologies exist and will be used during our work. As we explained above, the lack of existing training data does not allow us to provide a complete system for every possible relation type. We rely on statistical techniques providing evidence for the relation between entities with high recall and techniques that may increase the precision when training data exists.

On the other hand, even with training data available, the relations described in these datasets are vague and are not described in much detail. For instance, for protein interactions we can rarely depict the type of relation like modifying interaction vs. non-modifying interaction as we have shown in [124].

The Figure 5.5 shows the idea behind this extraction system[82][78]. The system generates a user query given a topic of interest and the top ranked documents are retrieved. Then the entities from the retrieved documents are extracted and ranked by the co-occurrence with the topic of interest. The co-occurrences only express the association between entities but not the type of association or the relation between entities. The labeling of the co-occurrences is described in Section 5.8.3.

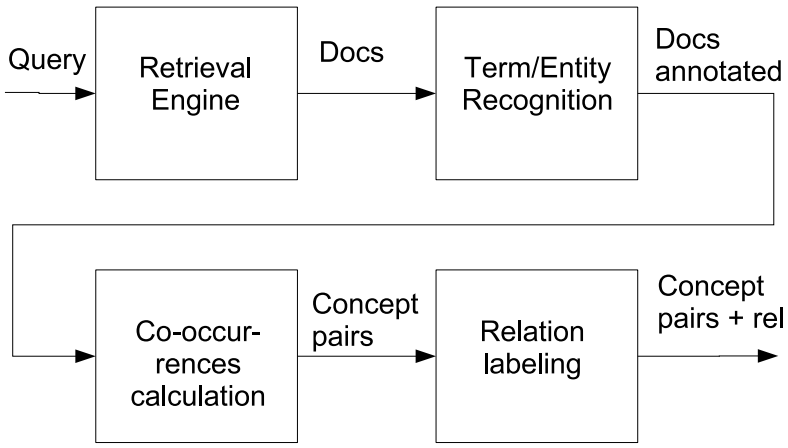


Figure 5.5: Concept relation based on co-occurrences and semantic labeling

We use dictionary look-up and we have seen that there are problems when different entities have the same surface form. By prioritizing the semantic en-

tities we solve some of the basic problems. A disambiguation algorithm based on the context has been implemented. The implementation is simple and looks for the concept with the highest contextual information. The contextual information is compared with a model of the concept based on its terminology and relations as expressed in the ontology. The idea behind the algorithm is based on the method by Agirre and Rigau[3]. The proteins and genes present an additional problem since the same term may denote several species. A heuristic has been used to select only proteins and genes from only the species of interest according to the dataset in use. This means considering either human or yeast proteins and genes.

The extraction delivers a large number of entities. We consider the entities that appear in conjunction with our entities of interest. To deliver the most interesting co-occurrences we have ranked them. We have considered several measurements to perform this ranking.

- Frequency: the most frequent co-occurrences will appear at the top ranks. In this measure we do not consider the individual distributions of $w_i$ and $w_j$. Concepts that have a high probability of occurrence in the collection will rank high as well even if they are not so informative. The other two measures try to avoid this issue.

$$Hits(w_i, w_j) \tag{5.5}$$

- Cohesion: this measure evaluates how linked the concepts are in the co-occurrence by considering how close to one of the elements of the pair is. It is sensitive to noise in elements with low frequency.

$$\frac{Hits(w_i, w_j)}{min(Hits(w_i), Hits(w_j))} \tag{5.6}$$

- Log-likelihood ratio: this measure considers not only the distribution of the pairs but also the individual distribution of the elements of a pair; performing a statistical test of independence of the elements.

$$-2log\lambda = 2log\frac{L(H_1)}{L(H_0)} \tag{5.7}$$

Examples of ranked PGNs for the disease Juvenile Idiopathic Arthritis and the protein and gene annotator are presented in Table 5.7. Co-occurrences just identify an association between two concepts; therefore we have to add a semantic label to this association. The semantic label is provided by the analysis of the sentences denoting the co-occurrences, as explained later.

| Rank | Frequency | Cohesion | LogL |
|---|---|---|---|
| 1 | rheumatoid factor | monocyte chemoattractant protein 1 | B27 |
| 2 | Etanercept | interferon regulatory factor 1 | Etanercept |
| 3 | ANA | ACTH | ANA |
| 4 | IL-6 | CD30 | IL-6 |
| 5 | TNF-alpha | IL-15 | CRP |
| 6 | TNF | HMG-1 | TNF-alpha |
| 7 | tumor necrosis factor | HMG-2 | TNF |
| 8 | B27 | IL1RN | DRB1 |
| 9 | interleukin-6 | CD5 | ferritin |
| 10 | DRB1 | CCR5 | IGF-I |
| 11 | CD4 | TNF beta | hemoglobin |
| 12 | Growth Hormone | IFA | CD4 |
| 13 | MIF | S100A12 | rheumatoid factor |
| 14 | HLA-DRB1 | MCP-1 | SAA |
| 15 | CRP | nuclear antigens | insulin |
| 16 | DEK | matrix metalloproteinase-3 | DR4 |
| 17 | Fas | Abs | DR5 |
| 18 | DR8 | TNF-a | axial |
| 19 | Insulin | haptoglobin | osteocalcin |
| 20 | HLA-A | CXCR3 | B19 |

Table 5.7: Co-occurrences PGN-JIA

### 5.8.3 Semantic labeling of co-occurrences

The co-occurrences presented above identify potential associations between concepts, but we still do not know which label can be given to a specific co-occurrence denoting a specific relation. Our proposal consists of identifying associations between concepts and then trying to determine the semantic label from a set of predefined types of relations.

We will perform the selection of semantic labels based on the classification of relevant sentences denoting the relation similar to[44]. The final decision is done by the selection of relevant sentences identified. Algorithm 5.1 is an example of this processing.

**Algorithm 5.1** Mapping co-occurrences to relation $r_i$ for concept $c$

```
for all co-occurrence c_j in C do
    count = 0
    for all sentence s_l in S_C do
        if s_l is classified as relevant to r_i then
            count = count + 1
        end if
    end for
    if f(count, S_C) > α add r_i(c_j) (set of related concepts r_i)
end for
```

The $f(count, S_C)$ expresses the relevant sentences identified for a relation and the parameter $\alpha$ encodes the desired level of confidence under which we consider that there is enough evidence to make the link between the sentence and the relation. Several corpora are available that we can use to cover some of the relations presented in Chapter 2 related to the ontology.

- The dataset for PGN-disease association is provided by the OMIM dataset by Craven.
- The dataset for protein-protein interaction for yeast is provided by MIP by Craven.
- The relation between GO terms and PGNs is provided by BioCreAtive I and the GOA (Gene Ontology Annotation). The Gene Ontology is divided in three branches: cellular location, molecular function and biological process.

In Table 5.9 we find the sentence classification performance for the different relation types. The results are based on configurations of several learning algorithms including decision trees (J48), support vector machines (SMV), k-nearest neighbors and Bayesian classifiers. As we can see in Table 5.9, SVMs perform better on the different datasets, as has already been shown in several categorization tasks [87]. This performance is not surprising since the number of features (Table 5.8) is quite large compared to the number of instances and

the features are sparse. Other algorithms like decision trees present stability problems and small changes in the training data produce different trees with poor generalization.

| Relation | No Sentences | Positive Sentences | No features |
|---|---|---|---|
| PGN-disease | 1793 | 856 | 4015 |
| PPI | 2501 | 1382 | 5594 |

Table 5.8: Sentence distributions

| Relation | Algorithm | Precision | Recall | F-measure |
|---|---|---|---|---|
| PGN-disease | J48 | 0.82 | 0.77 | 0.79 |
| PGN-disease | NB | 0.73 | 0.83 | 0.78 |
| PGN-disease | K-NN1 | **0.90** | 0.57 | 0.70 |
| PGN-disease | K-NN5 | 0.83 | 0.37 | 0.51 |
| PGN-disease | SVM | 0.87 | **0.83** | **0.85** |
| PPI | J48 | 0.69 | 0.73 | 0.71 |
| PPI | NB | 0.70 | 0.67 | 0.68 |
| PPI | SVM | **0.80** | **0.81** | **0.81** |
| PPI | K-NN1 | 0.80 | 0.57 | 0.66 |
| PPI | K-NN5 | 0.68 | 0.59 | 0.63 |

Table 5.9: Sentence categorization results

If we consider the co-occurrences presented above, the presented algorithm for semantic labeling of the relations selects the following co-occurrences as being related according to a specific relation.

The set of PGN concepts predicted by the approach can be summed up to: B27, etanercept, IL-6, MIF and TNF-alpha. In [78] we present the known PGNs for JIA (IL-6, MIF) and we already identified that TNF-alpha had some implications. B27 appears as hypothetical in the documents and etanercept appears as relevant for the treatment of JIA. In Table 5.7 we find the entities ranked by several statistics. We can see that the frequency provides all the PGNs mentioned in this analysis. We identify already well-known information in the documents, which is relevant for automatic processing of the literature, but we are not able to capture new information or make hypothetical statements. We assume that for the retrieval task we are dealing with, it is enough.

The relation types proposed can be extended to cover different parts of the domain not considered in our work. The available methods provide a performance rather weak and can make it more difficult to study the examples provided to the ontology refinement algorithm. On the other hand, the identification of relation types from text and the combination with the ontology refinement algorithm is proposed as future work.

## 5.9 Discussion

In this chapter we have presented IE that is relevant for our decision process since text sources are the only remaining source for our refinement algorithm. We have presented several approaches to perform the extraction of entities and have discussed which of the existing ones are more suitable for our work. We have contributed to the recognition of diseases based on the diseases present in the UMLS Metathesaurus and an approach to perform IE that relies on relevant and irrelevant sentences and co-occurrences. In the following chapter we discuss the experimental set up before we go to the results chapter.

# Chapter 6

# Experimentation

## 6.1 Experimental Strategy

In our work we intend to refine an ontology to improve IR performance. In order to quantify the improvement of our approach, if any, we need to define the experimental strategy that allows us, without any ambiguity, to evaluate our hypotheses. The improvement is measured by comparison of the performance before and after the refinement. This chapter is organized as follows. The following section introduces the evaluation strategy and typical performance measures in IR. Then we introduce the evaluation procedure required in our experiments and the different algorithms that we would like to evaluate and finally we introduce the datasets we have been working on.

### 6.1.1 Information Retrieval Evaluation

Evaluation implies the need to measure the performance of the system following a characteristic of the system under evaluation. There are several factors that can be considered in an IR system [159] like its speed or the document retrieval quality; being the latter the main focus of our work. In the following subsections we present the set up of a retrieval test collection and the measures used for the evaluation.

#### 6.1.1.1 IR test collection

An IR test collection usually follows the Cranfield tradition[32] being made of three components: a document collection, a set of information needs (e.g. queries) and relevance assessments which for a given information need relates the documents in the collection. This set up relies on several assumptions. The first one is that relevance can be approximated to a topic. Moreover, all documents are equally relevant, the relevance of a document is independent and that the user information need remains static during the test. The list of relevant documents is complete, even though this is difficult to obtain from

| | Relevant | Not Relevant | |
|---|---|---|---|
| Retrieved | $A \cap B$ | $NotA \cap B$ | B |
| Not Retrieved | $A \cap NotB$ | $NotA \cap NotB$ | Not B |
| | A | Not A | |

Table 6.1: Contingency table for the retrieved documents

current document collections. In large document collections it is difficult to find all the relevant documents in the collection; i.e. an assessor can not read all the documents in the collection to find all the relevant documents. In test collections the recall is always an approximation but this does not prevent to use the test collections following Cranfield tradition to be used to compare the performance of retrieval systems. Several techniques are provided in the literature to provide an accurate test document collection such as the pooling technique in TREC. In TREC the presence of several systems in the pooling mechanism has allowed to prepare reliable document collections[171].

In addition, Voorhees and Buckley[160] have shown that an IR test collections should contain a minimum of 25 queries for stability issues, with more queries there is not a significant difference. Some techniques propose to obtain a set of queries by using the log files of a retrieval system, so we can select the queries that are being most used in the system or by querying several domain experts.

#### 6.1.1.2 IR Performance Evaluation Measures

Performance measures allows the comparison of the retrieval performance quality of several approaches. [1] For each information need in the test collection, an information retrieval system retrieves documents from the collection. A retrieved document is categorized as relevant if it is in the list of relevant documents for the given information need; otherwise categorized as irrelevant. In this work this categorization is based on a IR benchmark. Once we have assigned the category to the documents we fill the contingency table 6.1. From this contingency table we calculate some of the evaluation measures more frequently used in IR.

Precision is the ratio of relevant documents in the set of retrieved documents and gives an idea of the quantity of noise in the retrieved set:

$$Precision = \frac{A \cap B}{B} \tag{6.1}$$

Recall is the proportion of relevant documents that are retrieved by the system:

[1] Even though some of this measures are used in information extraction and have already been presented in the previous chapter, they are included in this section to offer a clear presentation of the IR evaluation.

$$Recall = \frac{A \cap B}{A} \tag{6.2}$$

Fall-out is the proportion of irrelevant documents that are retrieved by the system [2]:

$$Fall - out = \frac{Not\_A \cap B}{Not\_A} \tag{6.3}$$

F-measure is the harmonic mean of precision and recall and provides a single measure to compare systems in comparison to precision and recall alone. The F-measure might be configured to give more importance to precision or recall by means of the alpha parameter.

$$F\alpha = \frac{(1+\alpha)Precision * Recall}{(\alpha * Precision + Recall)} \tag{6.4}$$

Typically alpha is set to 1; meaning that we do not prefer precision over recall and vice versa, even though it is becoming common to use different alpha values in large document collections:

$$F1 = \frac{2 * Precision * Recall}{(Precision + Recall)} \tag{6.5}$$

The measures presented above are useful to measure the performance of a Boolean system or a categorization system but are not useful, as such, to measure the performance of a ranking algorithm. In this case precision at different recall or rank levels can be used to measure and compare the ranking capabilities of the systems. A precision-recall curve is usually used to compare the ranking of two retrieval systems as presented in Figure 6.1.

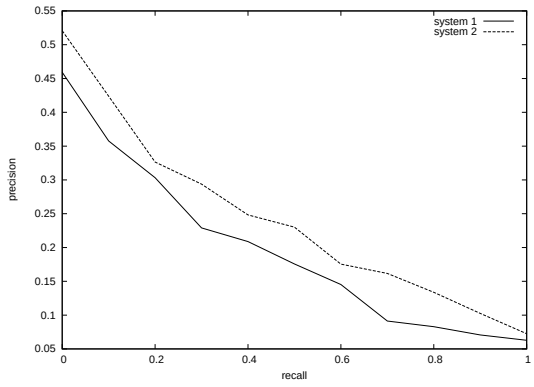


Figure 6.1: Precision-recall curve

[2] In our work we will not consider this measure since we will work with a certain proportion of top retrieved documents. The number of documents considered will contain a proportion of non-relevant documents that can be depreciated

From the precision at several recall levels we calculate the mean average precision (MAP); i.e. we calculate the average precision at recall levels $Recall_{ij}$ and we average this measure for the queries $Q_i$ in the test collection. This single measure is used to compare the ranking retrieval systems.

$$MAP = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{m_i}\sum_{j=1}^{m_i} Precision(Recall_{ij}) \tag{6.6}$$

Another measure is the R-Precision (RPr) or the precision at the number of relevant documents for the query. An R-precision of 1.0 means perfect retrieval. If R is higher than the number of documents retrieved then we consider that those documents are non-relevant.

Even though these are the performance measures most commonly used in the evaluation of IR systems, the ROC (Receiver operating characteristic)-curve[50] and the AUC area are being used more commonly on text categorization and they are becoming popular for ad-hoc IR. The ROC curve will compare the signal against the noise according to the different parameters of the retrieval system.

#### 6.1.1.3 Statistical Significance of the Results

When comparing retrieval systems we average the result of the different queries. Comparing two different approaches will mean to compare their performances. The average might be misleading since the good performance in some of the queries may hide the under performance of other queries, a statistical significance analysis is needed to find out if the difference is significant[171]. Parametric tests like the t-test should not be used since we cannot make the assumption of any parametric distribution of the behavior of the queries. In IR the Wilcoxon signed test[157] has shown to be suitable even though we have some guarantees, still risk errors of type I. We use a paired test on which randomized resampling is used by changing the sign of the difference and using this to obtain a distribution that is then compared to our result[167]. Randomized tests are not very popular in this field even though we may find them on other text mining tasks like IE. The results presented in Chapter 7 are tested statistically with this randomized test.

### 6.1.2 Experimental Plan

The plan for experimental procedure describes the steps that we are going to perform to run our refinement algorithm and to evaluate it against existing methods. The basic steps can be introduced as follows:

- Baseline preparation: consist of the language model approach and relevance based language model. The original query is either provided by the benchmark or is composed of the preferred terms as they appear in the relevant database. This sets the baseline comparison with the Query Expansion and Refinement using the original ontology.

- Result using the ontology based query: show the queries and maybe compare it to the relevance based language model.

- Then we perform the experiments with the ontology refinement. This involves considering the refinement of the lexicon, the lexicon linked to the concepts, the taxonomic relations and the non-taxonomic relations.

## 6.2 Experimental Datasets

We have already seen what is needed for an experimental benchmark in IR. IR benchmarks are expensive to obtain so we have to rely on existing resources. In the Biomedical Domain, some of the most relevant collections are: the OSHUMED collection, the cystic fibrosis collection [3] and the Genomic Track in TREC collections. The first two collections are rather old and the queries are diverse while we are aiming for a set of queries that are similar in the type of information they are after. In our work we have considered two benchmarks. The first collection, is based on the Genomic Trec 2005 and the second has been prepared from a database of protein-protein interactions (PPI) database. These collections use Medline, the largest abstract collection in the Biomedical Domain, as the source of the documents.

### 6.2.1 Genomics TREC 2005 collection

TREC (Text REtrieval Conference)[4] is funded by the National Institute of Standards and Technology (NIST) and has run for several years. In the last editions an interest on a Genomic Track has existed with different purposes that have moved from IR and text categorization to tasks similar to question answering. We have used the 2005 Genomic TREC collection because there is an interest on generic topics. This collection is made up of a subcollection of Medline, around 4M documents between the years 1999 and 2004, and a collection of 50 queries. The relevance assessments have been obtained using a pooling technique that is reviewed by several domain experts. Any document that is not in this pool is assumed to be non relevant. Zobel[171] has studied the reliability of the pooling method used in TREC. He found that the number of participants is large enough to ensure reliability of the comparison of different systems.

The queries in this benchmark are categorized in five groups defined by the generic topic template (GTT) [5]. We have selected a set of queries that relates 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 list of queries can be found in appendix A.

[3] http://people.ischool.berkeley.edu/ hearst/irbook/cfc.html
[4] http://trec.nist.gov
[5] http://ir.ohsu.edu/genomics/2005protocol.html

### 6.2.2 DIP protein-protein interaction

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 [6]; so this task is really relevant in the Biomedical Domain. Another new topic is the mining of gene regulation information from the literature for which less databases [7] and ontologies [8] exist to support it.

We would like to select relevant documents that indicate that two proteins are related and are useful for curation. The DIP database [9] deals with protein-protein interaction on yeast and has pointers to the source articles. In this database the yeast species has been more carefully curated than any other species. We have built a document collection out of it. We work again with a subset of Medline.

Queries are based on topical templates (similar to Genomics TREC 2005 queries): the interaction of protein X and protein Y. Appendix A shows the table of queries considered. In total 260 queries are prepared. The average number of documents 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 the documents collected for each one of the interacting proteins collected in the database for which an interaction has been collected. The main difference is that the analysis is done based on the full text of the documents and not based on the abstracts. This means that some documents have been annotated with many interactions. This 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 documents until September 2004 and the collection is much larger compared to the previous data set, about 15M Medline documents.

[6] http://biocreative.sourceforge.net/bc2ws/index.html
[7] http://www.oreganno.org/
[8] http://www.obofoundry.org/cgi-bin/detail.cgi?id=gene_regulation
[9] http://dip.doe-mbi.ucla.edu/

# Chapter 7

# Results

## 7.1 Introduction

This chapter presents the results of the experiments following the procedures described in the previous chapter. The results will allow us to measure the ability of the algorithm to improve an ontology and to compare the method to existing state of the art approaches in IR.

The preparation of the retrieval is based on the Lemur toolkit [1] which implements IR algorithms. As a general configuration, the stop words are defined by a standard list available from Lemur and the Krovetz stemmer is used. Specific configurations are presented for each one of the approaches compared.

The datasets used were introduced in the previous chapter and the documents are Medline abstracts in XML format. The fields *ArticleTitle* and *AbstractText* are processed to extract text which is used for indexing and retrieval. Even though more metadata is available we want to focus on the available text. This metadata includes the MeSH annotation and references to other sources but, even though, they can be used to increase recall they are a source of noise which are harmful to extract relevant conclusions.

This chapter is organized as follows: first we present the results based on standard IR used as baseline and already existing relevance models[96] as standard blind-feedback mechanism. Then we show the experiments obtained with the original ontology introduced in Chapter 2 based on the algorithm introduced in Chapter 3. The performance of the original baseline and the ontology based retrieval are presented. Finally we present the results of the refinement algorithm introduced in Chapter 4 in different scenarios like lexicon cleansing, ontology refinement and an analysis of the relations.

[1] http://www.lemurproject.org

### 7.1.1 Language Model

The language models have been presented in Chapter 3 and the main idea is to estimate the probability that the query is produced by the document language model and ranked the documents according to this probability as shown in the following formula.

$$P(Q|D) = \prod_{t \in Q} \lambda \frac{tf}{\sum tfi} + (1 - \lambda) P(t|C) \tag{7.1}$$

The parameter $\lambda$ relates the language model of the document and the background distribution.

### 7.1.2 Relevance Model

One of the research topics in IR is how to produce a better user query. In Chapter 3 we introduced the concept of pseudo-relevance feedback in the absence of explicit indication of relevance. The relevance models propose the estimation of a model based on the top retrieved documents for a given query in the language model approach. In this section we present the method to estimate a relevance model.

#### 7.1.2.1 Estimation of a relevance model

As we specified in Chapter 3, the estimation of the relevance model is based on the top ranked documents. The probability $P(w|R)$ is based on the probability of $w$ having seen the query words $q_1 \dots q_k$.

$$p(w|R) \approx P(w|q_1 \dots q_k) \tag{7.2}$$

This equation can be expressed as the joint probability of observing $w$ with the query words $q_1 \dots q_k$.

$$p(w|R) \approx \frac{P(w, q_1 \dots q_k)}{P(q_1 \dots q_k)} \tag{7.3}$$

Lavrenko presents two methods to estimate this probability. The first one assumes a sampling similar to the query words and the second assumes that $w$ and the query words are generated independently.

**Method 1: i.i.d. sampling** Let $C$ be the universe of unigrams that we use for sampling. Pick a distribution $D \in C$ with probability $P(D)$ and sample from it $k + 1$ times.

$$P(w, q_1 \dots q_k) = \sum_{D \in C} P(D) P(w, q_1 \dots q_k|D) \tag{7.4}$$

Sample independently and identically:

$$P(w, q_1 \dots q_k|D) = P(w|D) \prod_{i=1}^{k} P(q_i|D) \tag{7.5}$$

Express their joint probability as the product of the marginals:

$$P(w, q_1 \dots q_k) = \sum_{D \in C} P(D) P(w|D) \prod_{i=1}^{k} P(q_i|D) \tag{7.6}$$

**Method 2: conditional sampling** In the second method we consider w according to the prior $P(w)$. Then $k$ times pick a distribution $D_i \in C$ according to $P(D_i|w)$, sample query word $q_i$ from $D_i$ with probability $P(q_i|D_i)$. Keep independence $q_1 \dots q_k$ but keep their dependence on $w$:

$$P(w, q_1 \dots q_k) = P(w) \prod_{i=1}^{k} P(q_i|w) \tag{7.7}$$

Expectation over the universe C of unigrams:

$$P(q_i|w) = \sum_{D \in C} P(q_i|D_i) P(D_i|w) \tag{7.8}$$

Finally by substitution in the previous equations we get:

$$P(w, q_1 \dots q_k) = P(w) \prod_{i=1}^{k} (\sum_{D \in C} P(q_i|D_i) P(D_i|w)) \tag{7.9}$$

**Final Estimation Details** To comply with probability theory and to ensure proper additivity of the model, the prior $P(q_1 \dots q_k)$ is established as:

$$P(q_1 \dots q_k) = \sum_{w \in V} P(w, q_1 \dots q_k) \tag{7.10}$$

The word prior $P(w)$ is:

$$P(w) = \sum_{D \in C} P(w|D) P(D) \tag{7.11}$$

In the first method, Lavrenko arbitrarily used unigram distribution priors $P(D)$. In the second method the conditional probability of picking a distribution $M_i$ based on $w$ is:

$$P(D_i|w) = P(w|D_i) P(w) / P(D_i) \tag{7.12}$$

#### 7.1.2.2 Ranking with relevance models

Once we have the model, documents are ranked by the probabilistic ranking principle by Robertson.

$$\frac{P(D|R)}{P(D|N)} \sim \prod_{t \epsilon D} \frac{P(t|R)}{P(t|N)} \tag{7.13}$$

Another approach and equivalent for ranking purposes used in language modeling is the cross-entropy, in a similar way to the ontology retrieval approach:

$$CE(Q, D) = \sum_{t \in Q} P(t|Q) log(P(t|D)) \tag{7.14}$$

### 7.1.3 Ontology based Retrieval

In this work we have developed a query model that is used to link the ontology with the retrieval task. This ontology query model has been presented in Chapter 3. This method is compared against the other two in the following section.

The queries from the datasets presented in the previous chapter have been mapped to a conceptual representation; this means that the model will consider a mapping of the queries to concepts in our ontology. There are several parameters defined in the algorithm presented already in Chapter 3 that we just re-introduce in this section.

- Lambda for the documents, the same one as the specified for the document retrieval specified above. In the following section we study the effect of this lambda parameter in our proposal and the baseline models.
- Lambda for the words linked to terms in the ontology, empirical results do not provide better performance with values around 0.6.
- Lambda for concepts and related concepts. We have decided to apply a low value to provide more relevance to the query concepts. As this is a parameter needed for performance improvement we propose to consider a precise configuration within the experiments to tune it.
- The radius defines the concepts selected based on the selection of concepts from the ontology. Empirical evaluation is provided in the following sections.

### 7.1.4 Baseline results

This section presents the results based on state of the art techniques for IR and our ontological approach. The document model is based on the language model. It has been shown that the length of the documents[101] is a relevant parameter to be considered for the optimization of a retrieval system. Medline

contains citations from a large quantity of journals. These citations describe the meta-data of the article including the title and the abstract. Unfortunately, the documents are already quite small and for some of them only the title is present. In Figure 7.1 we find the cumulated distribution. The first years in the figure do not contain many mentions of abstracts while in recent years the number of citations with an abstract is catching up with the number of citations. Currently for half of the citations we find the abstract text. This means that is difficult to estimate the precise $\lambda$ value without empirical verification. The different baseline methods and our methods are compared using several values.

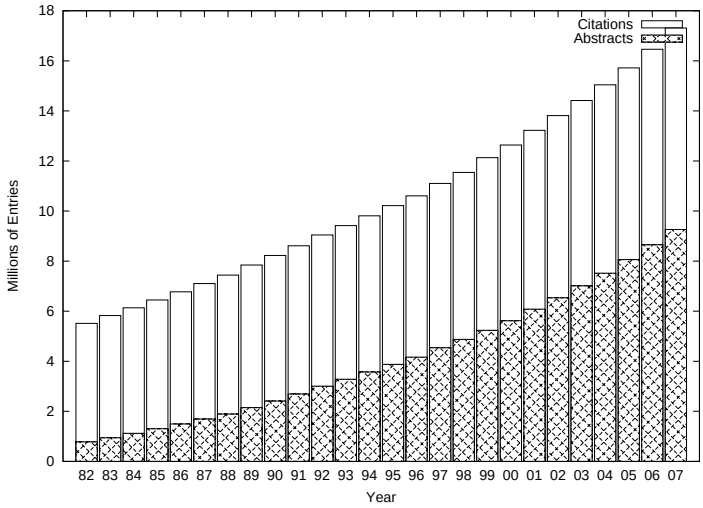


Figure 7.1: Citations against abstracts in Medline

In Tables 7.1 and 7.2 we can find the behavior observed for different $\lambda$ values for the different models. In this comparison we have considered the baseline language model based on JL smoothing (LM), the relevance models (RM) and the retrieval based on the ontology query model considering relations (Onto(r=1), the closest one) and without relations (Onto(r=0)). The results show that the optimal performance is obtained with higher values of lambda, this means that we give more relevance to the distribution of the words in the documents than the background distribution being used. The behavior is similar for medium and short queries but for long queries (as Onto(r=1)) we obtain better results with low $\lambda$ values.

In Tables 7.3 and 7.4 we show the models produced for different queries based on the relevance models and the initial ontology. As we can see, one difference is that the relevance models seem to favor general terms that seem to have a higher frequency in the documents like the terms *disease* and *protein*. On the other hand, the ontology query model presents some *redundant* terms that may hinder other terms from having a better estimation of their relevance. When $\lambda = 0$ only the word collection distribution is considered so all the documents get

| Lambda | LM | RM | Onto(r=1) | Onto(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 7.1: MAP performance of the models according to the lambda(PGN-disease)

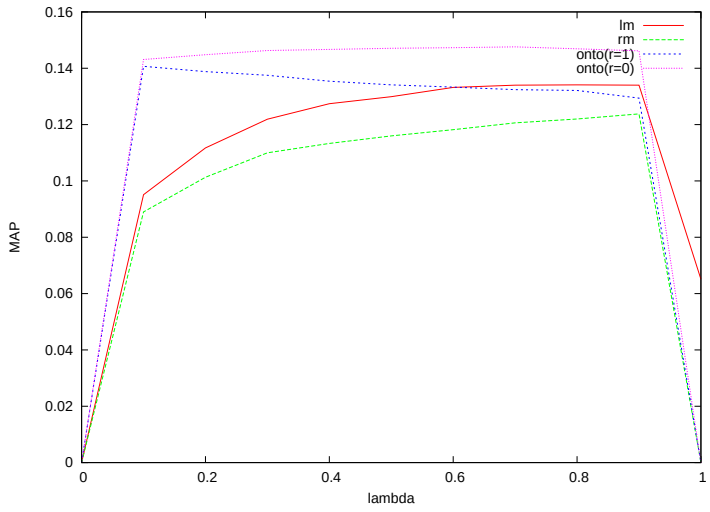


Figure 7.2: MAP for different lambda values (PGN-disease)

| Lambda | LM | RM | Onto(r=1) | Onto(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 7.2: MAP performance of the models according to lambda(PPI)

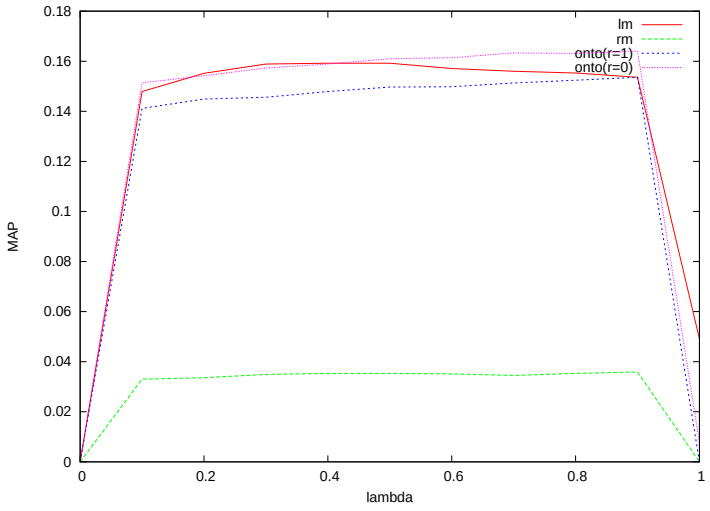


Figure 7.3: Lambda in the different approaches PPI

the same rank. The probability of retrieving the right documents is extremely low, so the probability is zero. On the other hand, when $\lambda = 1$ only the query model is considered. As we can see the results are very poor too. This shows the relevance of using a smoothed model.

| RM | $P(w_i\|R)$ | Ontology | $P(w_i\|\mathcal{C})$ |
|---|---|---|---|
| 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 7.3: Models for the query "IDE and Alzheimer disease" (10 top words)

| RM | $P(w_i\|R)$ | Ontology | $P(w_i\|\mathcal{C})$ |
|---|---|---|---|
| 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 7.4: Models for the proteins RAD51_YEAST and RAD52_YEAST (10 top words)

In Tables 7.5 and 7.6 and Figures 7.4 and 7.5 we can find the results obtained by using standard IR tools and the ontology approach introduced in Chapter 3. The language model used as baseline has better performance than the relevance model, as we have seen the relevance model gives higher probability to general terms. Other works on the TREC Genomics corpus have found, as well, that the usage of QE contributed negatively to the results.

The ontology model performs better than the language model but we obtain the best performance when no related concepts are considered; even for comparing results with different lambda values as we have seen above, there are several possible reasons for this. One of the reasons is that the relations are not relevant for the query and produce a *query drift*; this phenomena has already

been identified in IR [158]. In addition, it may happen that just some of the terms in the lexical entry linked to the concept are relevant. In this sense a cleaning of the terms should be done.

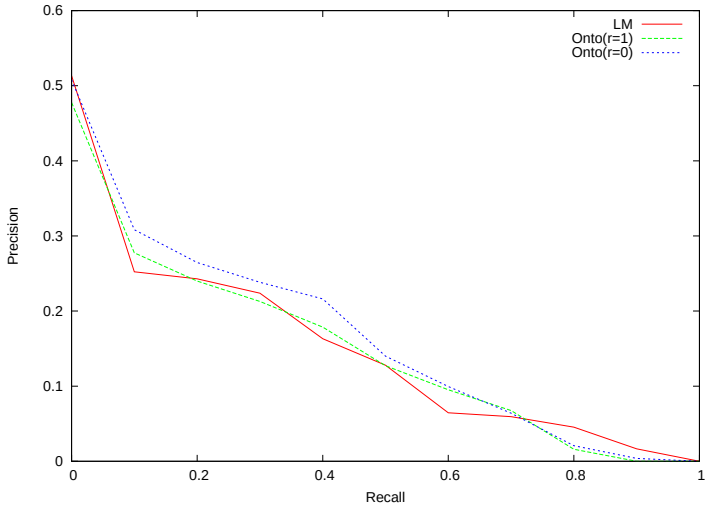


Figure 7.4: Precision-recall curve for LM and Ontology retrieval for PGN-disease

| TREC | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| LM | **1293/2117** | 0.1390 | **0.1968** |
| Onto(r=1) | 1115/2117 | 0.1311 | 0.1488 |
| Onto(r=0) | 1265/2117 | **0.1469** | 0.1680 |

Table 7.5: LM and Ontology retrieval results for PGN-disease

| PPI | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| LM | **456/642** | 0.1592 | 0.1221 |
| Onto(r=1) | 355/642 | 0.1536 | **0.1536** |
| Onto(r=0) | 395/642 | **0.1641** | 0.1291 |

Table 7.6: LM and Ontology retrieval results for PPI

### 7.1.5 Conclusions

The relevance models do not perform as well as expected and this is due to the fact that there is a bias toward one of the concepts that appear in the query as we can see in the example query where *Alzheimer disease* is more relevant than *IDE* and general collection terms like proteins. In the PPI set the performance of the ontology is better than the original query. This is due to the fact that

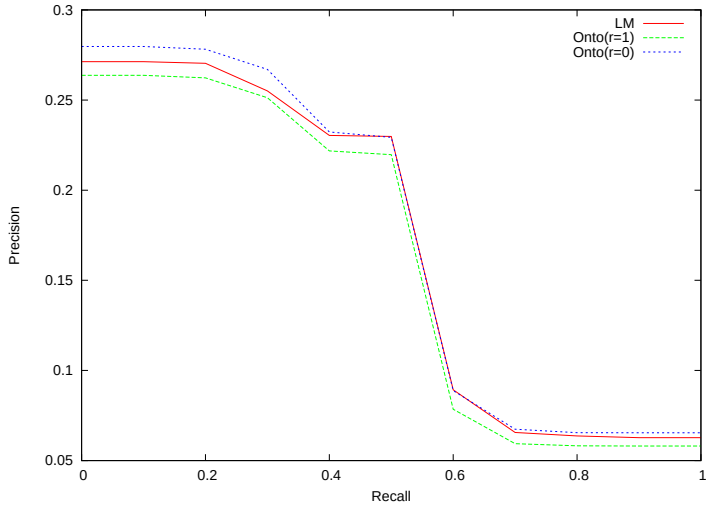


Figure 7.5: Precision-recall curve for LM and Ontology retrieval for PPI

species have several synonyms and not always the protein name is the best identifier for the protein. As we can see, the relevance models still have a bias for one concept or a different one that rank non-relevant documents first. Even so, the performance of OQM is within the mean performance of the systems presented in Genomics TREC 2005. The results point to several directions for ontology refinement.

The terms used are targeting irrelevant documents, false positives, so not appropriate, and cleansing is needed.

In the queries produced by the ontology model there are some terms that may be ambiguous or that are closer to a description than the term used in text to label the concept.

The lexicon in the ontology might not be complete so we miss documents since we cannot target them. Therefore, we can try to increase the recall by identifying potential synonyms not existing in the lexicon linked to the ontology.

We have seen that using related concepts worsens the results, there are two possible explanations that we can explore; either the lexicon is too noisy or the relations are not relevant for the topic of the query and *drift* the intention of the query.

The previous section has shown the behavior of the ontology retrieval presented on the two datasets. As we have seen several improvements can be proposed like cleaning the set of irrelevant terms and learning new terms but as well the usage of relevant concepts may have a relevant contribution. The different issues related to the lexicon and the concepts in the ontology are:

- Ambiguous and redundant terms are not appropriate for retrieval because they add noise to the produced query model.

- Some concepts are expressed using different terms, collecting those synonyms may reduce the false negatives rate.
- New concepts are needed for existing terms; some terms may be ambiguous but this ambiguity is not reflected in the ontology.
- Concepts related to the query concepts do not exist in the ontology.

These issues are considered in the decision process presented in Chapter 4 that produced ontology refinements. The refinement algorithm takes our ontology and proposes a refined version. As we discussed in Chapter 4, the algorithm is split into several steps on which some design decisions have to be taken according to the problem to be tackled.

In the next section we will study different procedures to perform the refinement of the ontology where the different issues will be discussed in detail. First we will study the cleansing of the lexicon of the query concepts, then we will study the possibility of increasing either the lexical entries in the lexicon or the relations between the concepts and several studies to clean the lexicon of the related concepts are proposed.

## 7.2 Lexicon Cleansing

As already expressed, one possible reason for the false positive rate 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*). We have to remember that we have considered an initial cleaning of the lexicon as shown in Chapter 2 while preparing the ontology related to common English terms and very frequent terms. The methods applied in this section work on this lexicon.

Several heuristics can be used like removing terms from the lexicon that cannot be found in Medline, this is query independent and may be considered as a first analysis. This will remove redundant terms that are of no use, so reducing the need for space and may reduce the noise introduced by these terms.

As we said above, some terms labeling a concept may not be used in some contexts. Without relevant documents we propose to evaluate if there is a preference for the terms given for the query based on the co-occurrence of terms labeling the query concepts and as well based on the related concepts defined in the ontology.

Finally, if we know some relevant documents we can estimate which terms are labeling the concepts that are used preferably. This is similar to already existing IR tasks where, based on a small number of relevant documents, we incorporate them in the retrieval mechanism to retrieve the remaining documents more effectively. The PGN-disease database contains many relevant documents per query, a random selection of relevant documents is used to select the terms having a high probability of appearing in the relevant documents. On the other

hand, the PPI set contains an average of two documents per query. In this case, it is not possible to do this analysis if we expect to produce a query that retrieves the remaining documents. As a baseline approach we have used relevance feedback to select relevant documents as the top-n documents retrieved by the original ontology query.

In these experiments we decide if a given term remains linked to a specific concept and no further probability is assigned that is not considered in the ontology query model presented in Chapter 3. This means that we consider the probability of 0 if the term is not considered or 1 if the term is linked to the lexical entry in the concept.

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

### 7.2.1 Term removal candidate selection

#### 7.2.1.1 Terms not in Medline

Some terms never appear in Medline but they 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 have not been removed. We have to remember that due to the length of Medline documents we expect to have specific terms more than descriptions of concepts.

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

In Tables 7.7 and 7.9 we compare the query models for the same query before and after removing these terms. We can see that many terms that looked like codes in databases have disappeared and that we clean the lexicon removing redundant terms.

#### 7.2.1.2 Co-occurrence of Medline Query Concept Terms

As mentioned above, some terms are used in contexts to denote a concept. Techniques in IR that use the notion of co-occurrence for term selection are not new. Similar to Cao et al.[25] the co-occurrences between terms in the corpus can be used as reference for the preference of the terms in use. Algorithm 7.1 estimates the co-occurrences of each of the terms of the query concepts. The frequency $freq_{t,c_j}$ is estimated querying the IR system where the term $t$ appears and any labeling $c_j$.

#### 7.2.1.3 Co-occurrence of Related Concepts in Medline

The preference of usage of terms denoting concepts can be discovered using the related concepts in the ontology and looking at their co-occurrences in text.

**Algorithm 7.1** 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_j in C do
4:       if c_i != c_j then
5:          for all term t in synset T labeling c_i do
6:             freq_t+ = freq_{t,c_j}
7:          end for
8:       end if
9:    end for
10:   for all term t in synset T labeling c_i do
11:      if freq_t < α then
12:         remove t from synset T
13:      end if
14:   end for
15: end for
```

Algorithm 7.2 depicts the behavior of the algorithm in pseudo-code.

**Algorithm 7.2** Ontology and collection cleansing

```
1: Given Concept c with terms t and relations r and corpus D
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 synset Tr labeling cr do
5:       for all t do
6:          if t and tr co-occur then
7:             freq_t+ = freq_{t,c_r}
8:          end if
9:       end for
10:   end for
11: end for
12: for all t do
13:    if freq_t < α then
14:       remove t from synset T
15:    end if
16: end for
```

#### 7.2.1.4 Refinement algorithm

In this section we want to use the refinement algorithm proposed in Chapter 4 to decide which terms that are linked to the concepts should be removed from the concepts. Before using the refinement algorithm we have to find the right set of parameters. The flaw extractor has to specify the following parameters:

**Term extraction** we extract from the documents the terms $T$ that label the concepts in the query.

**Term selection** the selection is done based on the probability that the terms appear in the relevant documents. The terms are ranked according to this probability and the top-n terms are selected. The probability, as explained above, is either estimated using a random selection of the relevant documents or by pseudo-relevance feedback.

During the analysis of the results we identity the set of terms to determine if there is a pattern of term preference based on this approach.

### 7.2.2 Lexicon Cleansing Results

In this section we compare the results obtained by the different approaches. In Tables 7.8 and 7.10 we can see the comparative result of the different methods. Compared to the baseline we can see an improvement independent of the technique being used. In these tables we find results for the following experiments: removal of terms not appearing in Medline (Corpus), co-occurrences of terms in each concept in the query (Coocur), co-occurrences terms from related concepts in the ontology (Onto), relevance feedback (RF), and results based on relevant documents (Rel). In the case of RF we have selected the top-5 documents per query while for the relevance we have considered 5 random documents per query. The results show that the cleaning of the lexicon is effective, meaning that the initial lexicon contains many terms that are not appropriate for IR even with the clean up presented in Chapter 2.

#### 7.2.2.1 PGN-disease cleaning results

The heuristics proposed for lexicon cleaning have a positive effect on the retrieval performance. As we see in Table 7.8 and Figure 7.6, the best method is based on the analysis of some of the relevant documents (statistical significance $p < 0.01$). We have compared the individual queries to better understand the behavior of the approach compared to the baseline ontology query model. As we can see in Figure 7.7 for many queries the improvement is quite large, in the query 119 terms like *mammary carcinoma* are removed leaving the term *breast cancer* which is more commonly used in relevant documents. The performance decreases significantly in query 93 where the cleaning leads to a significant decrease in document ranking, this is due to the removal of a relevant variant of the protein *drd4*.

In Table 7.7 we find that the 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* is used. This is a phenomena that we already found in[85] with the terms *cancer* and *neoplasm* which usually appear in the same synset while 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 has been not produced an interesting result. This is related as well to the result obtained in 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; considering that there is no notion of retrieval relevance.

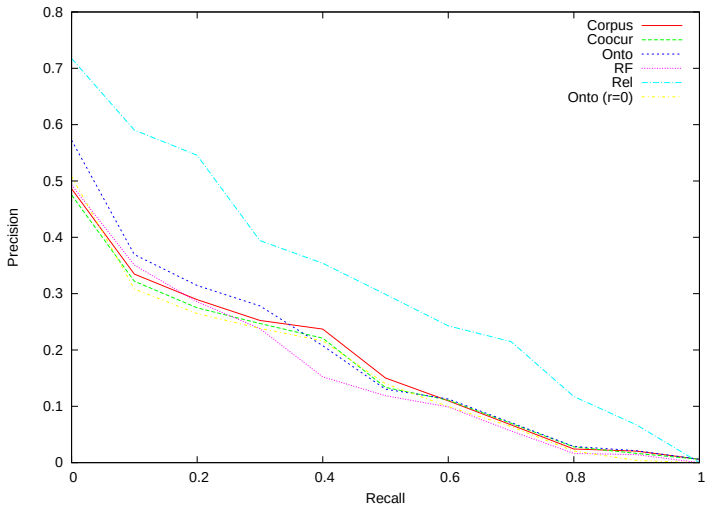


Figure 7.6: Precision-recall curve for lexicon cleansing for PGN-disease

#### 7.2.2.2 PPI cleaning results

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 use an explicit notion of relevance denoted by a user.

As we can see in Table 7.9, there are several terms that are targeted. Descriptive terms, e.g. *DNA repair protein RAD51*, are removed. The identifiers are no longer considered and specific terms without usage in co-occurrence with query concepts or ontology concepts are removed like *mut5*.

Results are presented in Table 7.10 and Figure 7.8. Relevance feedback has a lower performance than the other methods. We have already observed this behavior in the PGN-dataset and with the relevance models.

There are less relevant documents retrieved at the cut point of 1000 but

| 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 7.7: $P(w_i|\mathcal{C})$ for the query "IDE and Alzheimer disease" (10 top words)

| TREC | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Onto(r=0) | 1265/2117 | 0.1469 | 0.1680 |
| Corpus | 1345/2117 | 0.1635 | 0.1896 |
| Coocur | 1302/2117 | 0.1581 | 0.1923 |
| Onto | 1330/2117 | 0.1711 | 0.2151 |
| RF | 1031/2117 | 0.1524 | 0.1849 |
| Rel | **1483/2117** | **0.3080** | **0.3490** |

Table 7.8: Lexicon cleaning PGN-disease

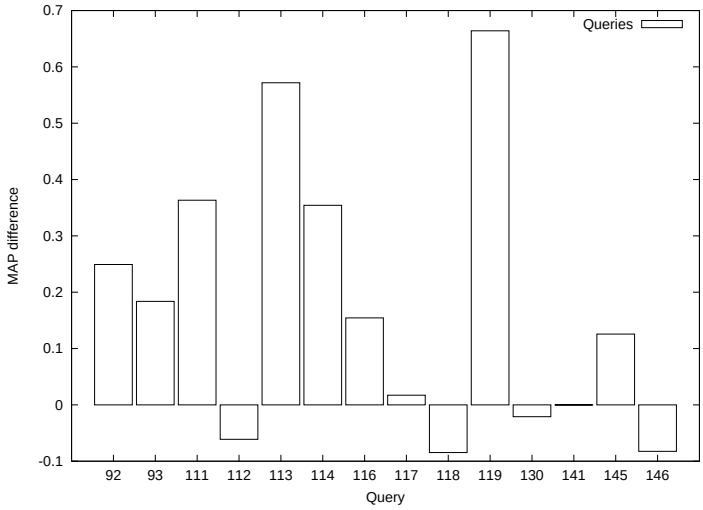


Figure 7.7: Comparison between original lexicon and relevance based cleaning

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 who 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 co-occurrences is similar. On the other hand, we find that the result is quite relevant since we can identify and target the terms that are not interesting for retrieval without a notion of relevance.

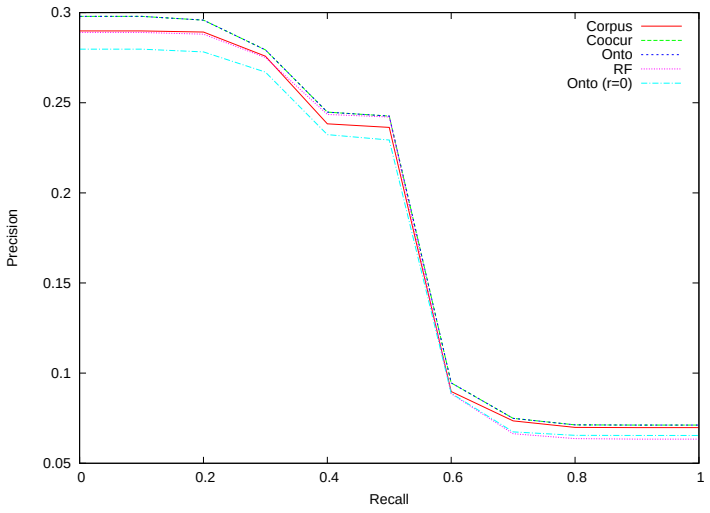


Figure 7.8: Precision-recall curve for lexicon cleansing for PPI

### 7.2.3 Conclusions

The different 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 the queries 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 the co-occurrence of related terms presents a better performance. Relevance feedback has a lower performance than the other methods. We have already observed this behavior in the PGN-dataset and PPI with the relevance models and in TREC Genomics this was already found.

The results obtained in this section allow us to remove part of the false positives but we still miss the false negatives delivered by the application and require decreasing the false positives. There are terms that are more difficult to remove without an indication of relevance that are close in meaning by composition.

| 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 | rad51 | 0.0467 |
| cerevisiae | 0.0667 | saccharomyce | 0.0583 | cerevisiae | 0.0583 | cerevisiae | 0.0583 | cerevisiae | 0.0583 |
| mut5 | 0.0533 | mut5 | 0.0467 | rad51 | 0.0467 | rad51 | 0.0467 | saccharomyce | 0.0583 |
| 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 7.9: $P(w_i|\mathcal{C})$ for the proteins RAD51_YEAST and RAD52_YEAST (10 top words)

| PPI | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Onto(r=0) | **395/642** | 0.1641 | 0.1291 |
| Corpus | 391/642 | 0.1693 | 0.1386 |
| Coocur | 376/642 | **0.1736** | **0.1406** |
| Onto | 376/642 | **0.1736** | **0.1406** |
| RF | 338/640 | 0.1673 | 0.1391 |

Table 7.10: Lexicon cleaning PPI

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 represented 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.

In the next section we work on improving the set of terms used to label the concepts and then we work on improving the relations between them. There are some terms that present ambiguity for which we will try to model the relevant context based on the query model. We will consider the possibility of improving the disambiguation based on the context in which the terms appear.

## 7.3 Ontology Refinement

The previous experiments have shown the effect of the lexicon cleansing applied to the original version of the lexicon and the link to the ontology. A similar study is offered in this section for ontology refinement; i.e. we are looking for specific knowledge required and to provide the mechanism to introduce this knowledge that has already been introduced in Chapter 4. This knowledge will consist basically of missing terms in the lexicon that may indicate missing concepts and missing relevant relations in the ontology required by the IR task.

The different components in the ontology refinement algorithm require a specific configuration within the scenario of our experiments. The flaw detector in the ontology refinement algorithm requires the identification of relevant features and the selection of these features. The decision process and the specific requirements concerning the link between IE and the ontology refinement and the specific configuration of the IE approaches were introduced in Chapter 5.

In the following subsections we introduce the specific configurations of the flaw detector and the different components of the ontology refinement algorithm. Finally we present the result and the main contribution of the different configurations of the algorithm with specific examples that represent the differences among these configurations.

### 7.3.1 Flaw Detector configuration

#### 7.3.1.1 Term extraction

As we have seen in Chapter 5, the identification of relevant terms from the documents is done based on syntactic analysis (shallow parsing) and based on NER for several entity types covered by our ontology. After the terms are selected and normalized to an ontology concept, we will identify the specific relations that will allow the query model to use them. Both approaches require different ways to deal with the identified terms. The NER approaches already disambiguate the existing terms labeling concepts in the ontology but just focus on known entities and do not provide new terms.

#### 7.3.1.2 Selection of Documents for Term Extraction

The extraction of terms is performed on set of documents from Medline. In the presence of relevant documents, the probability of a term occurring in the relevant set is considered. If negative example documents are provided, specific statistics can be estimated from the corpus to distinguish the set of the positive and the negative documents. In the absence of a relevance criterion, documents can be collected based on the top-n documents retrieved using the ontology query model query.

Another source might be to filter Medline based on co-occurrences. We identify sentences where both concepts appear. The selection of terms can be a simple frequency criterion or more complex statistics. Chapter 5 presents several formulas relevant to the co-occurrence analysis.

In this subsection we show an example query for which the different term extraction approaches are used to run on one query. These methods allow us to identify appropriate terms which are ranked according to the feature selection presented in Chapter 5. Tables 7.11 and 7.12 show the results of the first terms according to different samplings. The syntactic analysis provides different sets of terms that in many cases agree with the terms from the syntactic analysis and we can identify some differences. The NER proposes more specific terms and groups the different mentions of the concept in the same entry. The refinement experiments will allow us to evaluate the performance of the IE methods introduced in Chapter 5.

From this set of terms we will identify relevant terms from which we have to select accordingly to refine the ontology as we show in the following sections. One of the reasons for the relevant documents not being retrieved is that the ontology does not contain all the possible terms used to match the concepts in the query. We can see this clearly when we are not able to retrieve all the relevant documents for our queries. To solve this issue, the ontology can be expanded by adding new synonyms to existing concepts that might be required to relate new or already known concepts. Several techniques can be used to identify synonyms from the literature and have been introduced in Chapter 5. In addition, the source of ambiguities may come from the query terms that are not explicit enough to target the relevant documents. In this specific case, we

| Co-occurrences | | Relevance Feedback(8) | | Relevant | |
|---|---|---|---|---|---|
| APC | 182 | APC | 0.0135 | APC | 0.0104 |
| colon cancer | 94 | FAP | 0.0104 | beta-catenin | 0.0063 |
| mutations | 67 | colon | 0.0083 | treatment | 0.0062 |
| adenomatous poliposis coli | 31 | mice | 0.0083 | mutations | 0.0052 |
| colon cancer cells | 27 | Apc | 0.0073 | gamma-catenin | 0.0052 |
| beta-catenin | 25 | Min | 0.0073 | sulindac sulphide | 0.0052 |
| familial adenomatous poliposis coli | 21 | gene | 0.0052 | colon | 0.0041 |
| FAP | 17 | familial adenomatous polyposis | 0.0042 | 10 microM | 0.0041 |
| colon cancers | 16 | mutations | 0.0042 | rectal cancer | 0.0041 |
| loss | 15 | mouse | 0.0031 | regression | 0.0031 |

Table 7.11: Term extraction and syntactic analysis “APC AND Colon cancer”

| Co-occurrences | | Relevance Feedback | | Relevant | |
|---|---|---|---|---|---|
| APC | 370 | apc | 0.0027 | apc | 0.0250 |
| colon cancer | 235 | cancer | 0.0010 | beta-catenin | 0.0229 |
| beta-catenin | 92 | mice | 0.0010 | sulindac | 0.0170 |
| adenomatous poliposis coli | 89 | adenomatous poliposis coli | 0.0083 | cancers | 0.0156 |
| protein | 46 | protein | 0.0083 | cancer | 0.0156 |
| tumor | 45 | mouse | 0.0083 | gamma-catenin | 0.0125 |
| familial adenomatous polyposis | 24 | familial adenomatous poliposis | 0.0073 | sulphide | 0.0083 |
| cancer | 20 | apc(min | 0.0073 | tumor | 0.0073 |
| cancers | 17 | tumor | 0.0062 | sulphone | 0.0073 |
| colonic | 15 | colonic | 0.0041 | celecoxib | 0.0073 |

Table 7.12: Term extraction and IE and “APC AND Colon cancer”

would like to target the related terms that may be used to rank the documents according to relevance. The IE techniques are defined by the ontology refinement needs introduced in Chapter 4. In the following section we show the results of the refinement algorithm applied to our datasets.

In Tables 7.11 and 7.12 we find the ranking used to prioritize the terms. This ranking is used to select the terms that will run the decision process to determine their integration with the ontology and the query concepts. Examples of the outcome of the decision process can be found in Tables 7.17 and 7.18.

### 7.3.2 Refinement Algorithm results

We first show the results without applying the lexicon cleaning approach presented in the previous section. Later we will find the results on which the refinement cleaning algorithm has been applied.

#### 7.3.2.1 Refinement PGN-disease results

In Tables 7.13 and 7.14 we provide the results based on the identification of new terms and the identification of named entities (existing concepts). As we can see in Figure 7.14, the usage of new terms has a statistically significant lower ranking than using known concepts.

On the other hand, we see that the results obtained using relevant documents do not have the same effect as we found during the lexicon cleansing. The main reason for this behavior is that while in the lexicon cleansing we were trying to identify the most relevant terms from the set of terms linked to the concept, those seem to appear equally distributed. The terms related to the query terms seem to appear sparse along the relevant documents.

This means as well that if we calculate statistics over the terms appearing in the relevant documents we need a considerable number of relevant documents so the measurement does not become biased by the set presented to the algorithm.

We find as well that the results for the three algorithms for document selection offer a similar performance. This means that filtering the terms by means of the decision process based on IE is useful as we can find in the relevance feedback results. In addition, the performance of the co-occurrences means that it is possible to improve the retrieval without any relevance indication.

| TREC | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Onto(r=0) | 1265/2117 | 0.1469 | 0.1680 |
| Coocur | 1289/2117 | 0.1517 | 0.1788 |
| RF | 1291/2117 | 0.1518 | 0.1725 |
| Rel | **1293/2117** | 0.1516 | 0.1730 |

Table 7.13: Refinement using syntactic analysis PGN-disease

Now we show the results using the refinement algorithm presented above with the original ontology and the cleaned version of the lexicon using the set

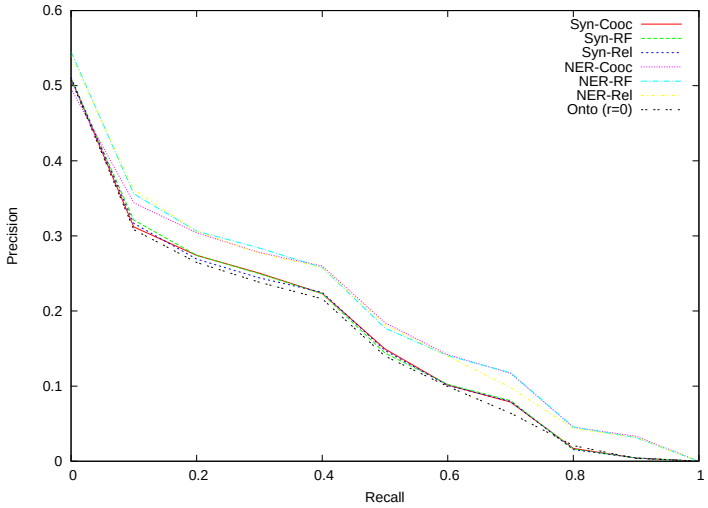


Figure 7.9: Precision-recall curve refinement PGN-disease

| TREC | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Onto(r=0) | 1265/2117 | 0.1469 | 0.1680 |
| Coocur | 1263/2117 | 0.1731 | 0.1944 |
| RF | **1287/2117** | 0.1755 | 0.1948 |
| Rel | 1283/2117 | **0.1766** | **0.1957** |

Table 7.14: Refinement using NER PGN-disease

of relevant documents. We have studied the impact of the refinement algorithm without any assumption about the transformations performed on the ontology in the previous sections.

We present now a second set of experiments to study the performance of the refinement algorithm when we have performed a cleaning of the lexicon using the best result obtained in the lexicon cleansing section. In the lexicon cleansing process the lexicon of the query concepts was revised and only the terms with high probability of occurrence where left. Now we would like to relate the query concepts with new or already existing concepts. In Tables 7.15 and 7.16 we show the results for the lexicon cleansing.

As we can see in Figure 7.10, in the system based on NER, relevant documents obtain a larger improvement. On the other hand, the refinement obtained with the feedback mechanism does not seem to perform well obtaining worse results than the baseline.

This is due to the fact that the quality of the top return documents is not high enough to provide features that allow us to obtain an improved ontology.

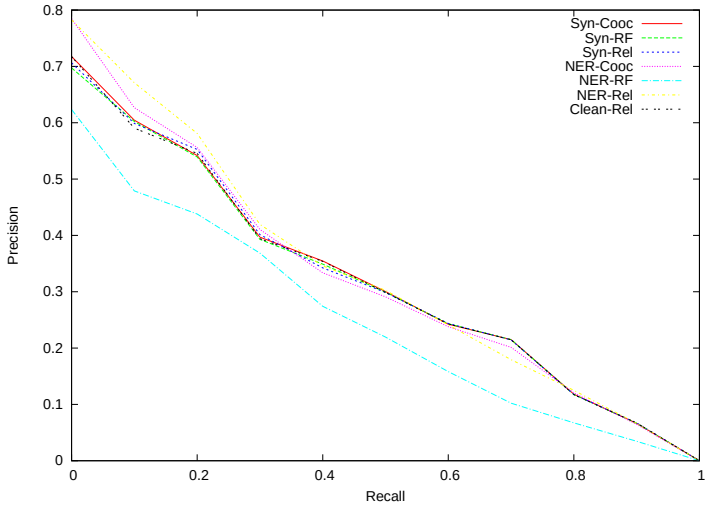


Figure 7.10: Precision-recall curve refinement and cleaned lexicon PGN-disease

| TREC | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Rel-base | 1483/2117 | **0.3080** | **0.3490** |
| Coocur | 1484/2117 | 0.3079 | 0.3467 |
| RF | 1485/2117 | 0.3051 | 0.3466 |
| Rel | 1485/2117 | 0.3068 | 0.3449 |

Table 7.15: Refinement relevant clean terms PGN-disease

In Tables 7.17 and 7.18 we find two example queries and the proposed refinements. In the first table we identify relations between related proteins and

| TREC | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Rel-base | 1483/2117 | 0.3080 | 0.3490 |
| Coocur | **1501/2117** | 0.3122 | 0.3537 |
| RF | 1464/2117 | 0.2375 | 0.2797 |
| Rel | 1492/2117 | **0.3198** | **0.3602** |

Table 7.16: Refinement relevant clean concepts PGN-disease

related genes and related diseases. We identify the disease *colorectal neoplasm* linked to the *APC gene* which as well is related to *colon cancer* that is the disease in the query. There is a clear relation based on hyponymy and it is as well represented in the ontology. We are able to identify a common gene relevant for both diseases.

In the second table we identify several interacting proteins. It is interesting as well to find that the disease *ovarian cancer* is related to one of the genes. This disease is a more specific than *cancer*.

The two queries present commonalities like the link to interacting proteins but as well differences since, on one hand, we can find more general concepts and, on the other, we can find more precise concepts.

We find as well an increase in the number of relevant retrieved documents. This means that based the related terms we are able to target documents that do not contain the concepts in the documents.

| Operation | Concept 1 | Concept 2 |
|---|---|---|
| AddR(6,226525,235082) | APC protein | beta-catenin |
| AddR(5,4003110,235082) | colon cancer | beta-catenin |
| AddR(5,4003110,1189772) | colon cancer | CDX-2 |
| AddR(5,4011125,226525) | adenomatous poliposis coli | APC protein |
| AddR(5,4015179,226525) | colorectal neoplasms | APC protein |

Table 7.17: Refinement for query "APC and colon cancer"

| Operation | Concept 1 | Concept 2 |
|---|---|---|
| AddR(6,237775,1190024) | BRCA1 | BRDA1 |
| AddR(6,259339,1190024) | Ubiquitin | BRDA1 |
| AddR(5,4010051,237775) | ovarian cancer | BRCA1 |
| AddR(5,4009369,1190024) | cancer | BRDA1 |
| AddR(5,4009369,257493) | cancer | UBE2N |

Table 7.18: Refinement for query "BRCA1 and ubiquitin AND cancer"

On the other hand, it is not possible to identify new relevant terms or concepts that are not already in the ontology. This is because the lexicon and the ontology already contains information about well known concepts that our algorithm is able to find. Future work should investigate the possibility of look-

ing for terms that are not so obvious to common statistical estimation but the performance of retrieval might be limited.

#### 7.3.2.2 Refinement PPI results

In Tables 7.19 and 7.20 and Figure 7.11 we show the results of applying the refinement of the ontology using several heuristics. In these experiments we can not use explicit feedback since the number of relevant documents per query is small and we rely only on some heuristics. The results do not improve over the baseline. This baseline is the result of the experiments using the lexicon cleaning based on co-occurrences. The result is interesting because the relations between the concepts are correct and relate the protein with its function. This means that the reformulation of the query has to consider specific knowledge that we cannot identify using the heuristics presented in this section.

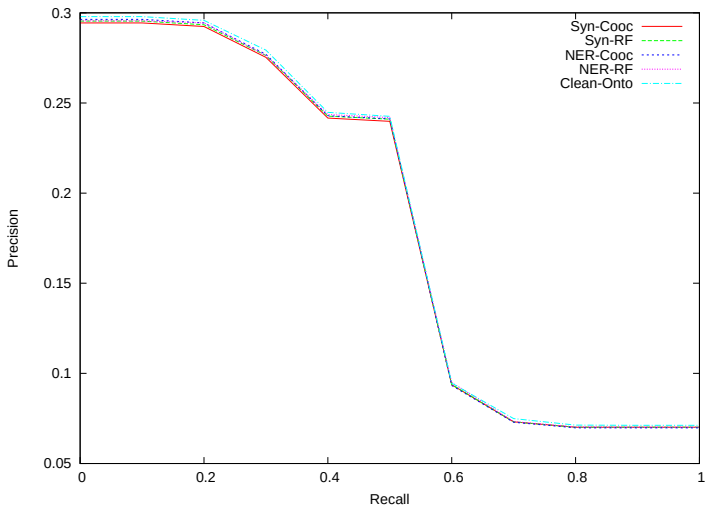


Figure 7.11: Precision-recall curve refinement PPI

| PPI | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Coocur | 373/642 | 0.1716 | 0.1399 |
| RF | 374/642 | 0.1722 | 0.1419 |

Table 7.19: Refinement using syntactic analysis PPI

### 7.3.3 Refinement of the Relations

In the previous sections we have studied the extraction of facts that is mainly concerned with the concepts of the ontology (terms and relations) and we have shown how this knowledge can help to improve the retrieval performance. But

| PPI | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Coocur | 374/642 | 0.1723 | 0.1451 |
| RF | 374/642 | 0.1726 | 0.1412 |

Table 7.20: Refinement using NER PPI

retrieved documents may discuss different topics that may not be of interest to the query that may not be captured by the previous method. This means that the concepts appear in a document but the document does not deal with the explicit relation of the query concepts.

We are interested in identifying terms to enrich the query provided by the relations/topics. This requires a modification of our ontology query model to integrate the terms denoting the relation in documents. The ontology query model is updated to consider the terms characterizing the relation $R$ as part of the term in the model. The relation $R$ is combined with the conceptual selection $C$. A linear combination of the related terms linked to the concepts and the relation terms is used:

$$P(w_i|C,R) = \alpha P_{CM}(w_i|C) + \beta P_R(w_i|C) + \gamma P_{Rel}(w_i|R) \tag{7.15}$$

$$\alpha + \beta + \gamma = 1 \tag{7.16}$$

The probability $P_{Rel}(w_i|R)$ depends only on the terminology applied to this relation and the others in the ontology. In the case of a richer relation ontology the probability would as well consider the occurrence of the terms in the other relations. An example can be the UMLS semantic network, but we find that the relation *interacts with* has as verb *interact* to denote the relation and we have found more terms linked to the relation, as we see later on.

Targeting documents that specifically discuss these relations is relevant. The extraction of terms that are denoting these relations is developed in this section. This derives into topic detection which may be linked into the ontology by refining the existing relations. Even though this is very challenging it is out of the scope of this work.

In order to learn the relevant topic features we intend to generalize over a selection of queries the appropriate information that may be used to improve retrieval. The documents are already annotated as relevant to the query in the dataset. The analysis of relevant and irrelevant documents for a set of documents retrieved for a query may allow identifying terms that are indicators of the relation.

The problem could be expressed as well as an information categorization problem and we would like to evaluate this against our ontology refinement approach. The queries will be split into a set of training and test set. The evaluation of the retrieval will then be done on a subset and an average of the performance will be presented. The set of queries for this dataset has been split in training and test using 5 times 2 fold cross validation[43].

The training set is based on the training queries that are used to retrieve the top-50 documents for each query. The positive documents are the documents relevant for the query while the non-relevant documents are considered as negatives. The documents with the label are placed in the same set. Since the number of negative documents overwhelms the number of positive ones, a random selection of negative documents is done to balance both classes.

In the refinement approach, the features of the dataset are ranked according to the probability of relevance. We look for features that are indicators of relations in the documents; both verbal forms and nominalized forms that are represented as verb and noun phrases. The presence of negative documents allows us to use statistics that rely on the presence of features that are interesting to discriminate both sets. Information Gain (IG) measures the reduction in entropy and is chosen to rank the features. The entropy of a random variable X $H(X)$ indicates the smallest number of bits needed on average to send a message from a stream of symbols drawn from $X$.

$$H(X) = -\sum_{i=1}^{m} p_i \log_2 p_i \tag{7.17}$$

Then, the information gain from the training examples $T$ for the attribute $a$ considers the entropy of the training examples and the conditional entropy of the attribute. In the estimation we will consider the different values of the class attribute to determine information gain of the attribute.

$$IG(T|a) = H(T) - H(T|a) \tag{7.18}$$

A baseline approach based on the best performing classifier is done to compare the result with the ontology query model. The classifier is applied on the list of the retrieved set for a given query and is used to boost documents that are classified as relevant. These documents are boosted to the top of the retrieved list keeping the original rank among them. The idea of boosting is similar to the work of Ruch and Geissbhler[138] that combines a traditional vector space model result and a rule based system. In the following sections we show the result of the analysis.

#### 7.3.3.1 PGN-disease dataset results

The configuration for the PGN-disease dataset is based on the result obtained from the relevance cleaning and refinement presented before. From this set, the positive and negative documents are obtained and the dataset for the classification and feature selection processes is obtained. In Table 7.21 we can see the result of selecting the result obtained for the classifiers. The documents have been tokenized, the stop words have been removed and no stemming is applied. The Naive Bayes classifier obtains the best results in the cross validation analysis. The result is very poor and the result for document boosting may be limited.

| Algorithm | Precision | Recall | F-measure |
|---|---|---|---|
| J48 | 0.4429 | 0.4055 | 0.3992 |
| NB | 0.5624 | 0.6457 | 0.5934 |
| SVM | 0.6235 | 0.3452 | 0.4424 |
| K-NN1 | 0.2231 | 0.0254 | 0.0454 |

Table 7.21: Document categorization results for PGN

In Table 7.22 we find the most relevant features in every fold. On the other hand, these features do not seem to be related to the relation that we are considering. We have to remember that this dataset contains a small number of queries and the relevant documents vary according to the query.

| Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 |
|---|---|---|---|---|
| disease | gstm1 | gstm1 | apc | disease |
| mutation | polymorphisms | genotype | polyposis | transforming |
| 129 | glutathione | null | adenomatous | mutations |
| onset | 0 | polymorphisms | coli | major |
| mutations | genotype | 0 | gstm1 | familial |
| alzheimer | null | cases | gene | alzheimer |
| lines | study | gluthatione | familial | genetic |
| familial | cases | genes | colon | onset |
| bard1 | genotypes | controls | polymorphisms | patients |
| allele | transferase | allele | 0 | linked |

Table 7.22: Feature selection for PGN-disease

In Table 7.23 and Figure 7.12 we compare the baseline based on the refinement algorithm and the results obtained with the categorization. As we can see, the best results are obtained with the baseline algorithm. This was expected since the classifiers presented above have a poor performance that seems to be due to the training set which does not allow finding a model that discards documents about the role of the PGN in the disease. As a set of terms relevant to the relation has not been found we have not shown results based on the ontology query model. A larger dataset could provide a better performance.

| TREC | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Refinement-base | 747.2/1093.6 | 0.3208 | 0.3608 |
| Categorization | 747.2/1093.6 | 0.2604 | 0.3033 |

Table 7.23: Refinement cleaning and categorization for PGN-disease

#### 7.3.3.2 PPI dataset results

Table 7.24 shows the results of documents categorization. In contrast to the results for the PGN-disease data set we see that the performance of the classifiers

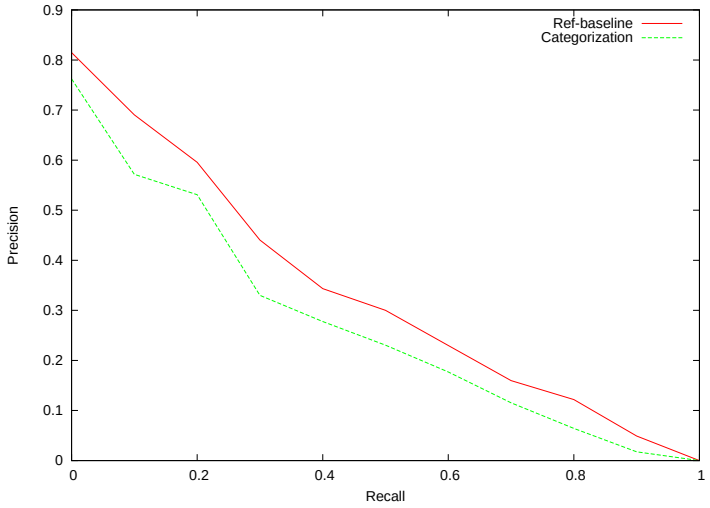


Figure 7.12: Precision-recall curve relation refinement PGN

is much better. Again, the documents are tokenized, stopwords are removed and tokens are not stemmed. The SVM obtains the best performance in the cross-validation analysis.

| Algorithm | Precision | Recall | F-measure |
|---|---|---|---|
| J48 | 0.7413 | 0.7828 | 0.7605 |
| NB | 0.6827 | 0.9906 | 0.8078 |
| SVM | 0.7953 | 0.8483 | 0.8202 |
| K-NN1 | 0.9611 | 0.1558 | 0.2663 |

Table 7.24: Document categorization results for PPI

In Table 7.25 we identify the tokens that are ranked by information gain. In contrast to the PGN-disease set, the tokens are more homogeneous in the different sets. From the list of terms, there are terms that clearly denote an interaction like *interaction, binding, complex and hybrid*, terms that are related to experiments done to verify the interaction between proteins. These terms have been found relevant in a similar study by Marcotte et al.[104] and Cohen et al. [33]. There are less obvious terms like *association* that have been found relevant in Rebholz et al.[124]. Almost all these features seem to be linked to the positive class, meaning as well that the features denoting other topics are more difficult to identify and the sub-topic analysis for this set may require more data than we have used.

In Table 7.26 and Figure 7.13 we present the result comparing the baseline methods with the modified ontology query model. In this case, the baseline methods are the co-occurrences based on cleaning and refinement and the classifier approach based on this method. We have used the trained SVM model to

| Fold 1 | Fold 2 | Fold 3 | Fold 4 | Fold 5 |
|---|---|---|---|---|
| protein | interaction | interaction | protein | interaction |
| essential | protein | complex | interaction | essential |
| interaction | binding | protein | interactions | required |
| hybrid | hybrid | binding | binding | protein |
| proteins | proteins | hybrid | proteins | proteins |
| interacts | complex | interacts | association | complex |
| complex | vitro | required | vivo | binding |
| binding | vivo | vivo | domain | vivo |
| show | essential | association | required | hybrid |
| domain | required | proteins | complex | vitro |

Table 7.25: Feature selection for PPI

perform the boosting of documents due to its performance in document categorization as found in Table 7.24.

As we can see in Table 7.26 and Figure 7.13, both approaches perform better than the baseline (statistically significant $p < 0.01$). Boosting based on the text categorizer provides a better performance. This means that there are specific arrangements that the model produced by the SVM captures better than the ontology query model.

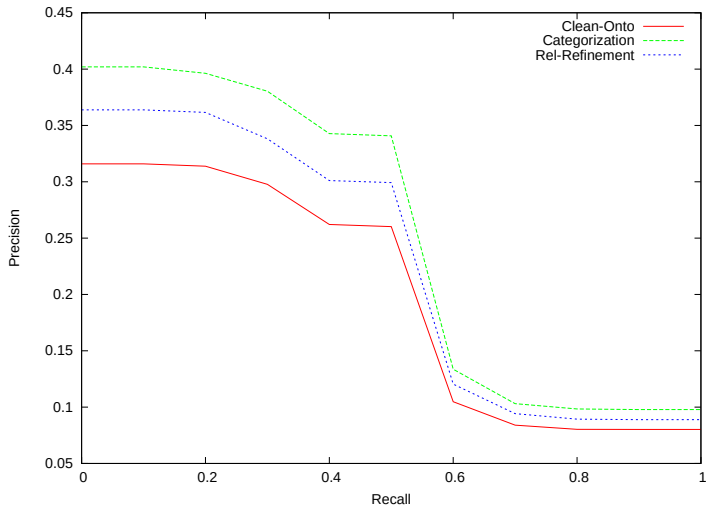


Figure 7.13: Precision-recall curve relation refinement PPI

### 7.3.4 Conclusions

The results show that the refinement of the ontology applied to IR is interesting. As we have seen, it is possible to identify missing knowledge in the ontology that can be used in IR. The ontological content used to improve a user query is

| PPI | Rel. Retrieved | MAP | R Precision |
|---|---|---|---|
| Co-occurr-base | 189.2/317.2 | 0.1873 | 0.1534 |
| Categorization | 189.2/317.2 | **0.2387** | **0.1993** |
| Refinement | **199.2/317.2** | 0.2140 | 0.1926 |

Table 7.26: Co-occurrence, categorization and refinement for PPI

query dependent meaning that a selection of the represented knowledge linked to a concept has to be done to avoid a query drift. This has already been found in the literature [158] and in this work we have proposed a method to revise the ontology to answer a given query.

We have shown that the lexicon in the existing resources contains terms that are not useful for retrieval since there is a preferred set of terms usually occurring in text.

The refinement based on relevance, using relevant documents, has proved to be effective. The selection of terms based on co-occurrences has been very effective and in some cases not much different than using relevant documents. On the other hand, the refinement based on pseudo-relevance feedback has been effective in some cases but not in others.

The initial experiments have shown, as well, that the ontology contains relations that are not relevant for retrieval. In addition, it seems that the related terms need to be closely related to the query concepts. We have been able to combine IR and IE. The extraction system that we have prepared has produced interesting results. As we have seen, the most useful relations in the PGN-disease set have been the ones relating PGNs to diseases and the protein interactions linked to the related PGNs appearing in the query that have a close relation to the disease. On the other hand, even though the IE predicted correct relations in the PPI dataset, the relation types do not contribute to improve the document retrieval task.

Finally, we have investigated into the topic of the set of queries of the dataset and have tried to populate the relations with relevant terms to identify the relation. We have seen that with enough training data we can effectively target relevant documents and identify terms that are typically used to identify the relations. We have targeted the topic of the queries but other topics could have been detected that are not related to the query and are being used to discard some of the non-relevant documents. There is some work focused on topic detection that is interesting to identify this topical set[14] and proposals exist to be integrated in the language models[161].

# Chapter 8

# Conclusions

## 8.1 Summary of the Results

In this work we have as objective to show if a domain ontology is useful for IR and if the ontology can be improved in view of the IR task. We have shown that this is possible and have determined the cases where it works. The results have required to further develop different aspects that we highlight in this section.

Ontologies are explicit specializations of a conceptualization and present, by default, no explicit link with the textual source. This link can be done by labeling the concepts with entries in a lexicon. Even though this has already been implemented in different systems, we have presented a model that combines lexicons which can be used in different ontologies. This link has been provided by existing resources where the terminology for our entities of interest has been extracted. Efficient reuse of existing resources has allowed us to develop an initial combination between the ontology and the lexicon.

Lexical ontologies and lexical resources have been used in the literature in different ways. We have proposed an approach to integrate our ontology and lexicon into the language model framework by providing a query model based on the ontology and its lexicon. The experiments have shown that we can provide better results than the results obtained by the language model approach for the datasets used as gold standard. These experiments have shown as well that our dataset has some specific issues, since citations are small since the documents contain only the abstract and the title. In addition, half of Medline citations only contain the abstract as we have shown. For the document retrieval it is more effective to specify the precise terms rather than having a query with context terms.

We have identified some expected results like the lower performance obtained when we were using the related terms of the concepts. This is due to the size of the documents, where very specific terms related to the query are required. The obtained results are in accordance with existing results identified by Voorhees[158] and we have provided a mechanism for a better integration of

terms labeling the concepts in the ontology and the IR problem.

As said above, the query model that we have developed depends on the ontology. This means that changes in the ontology or the lexicon produce a change in the model and have an impact on retrieval. This means that we can measure the suitability of the ontology and the changes applied to it against the retrieval task. We have proposed an algorithm to revise the ontology and the lexicon under this assumption. With this algorithm we have targeted several issues that may be hindering the retrieval of documents.

This algorithm analyzes the feedback, either provided by the user or by pseudo-relevance feedback, and produces possible changes to the ontology. The algorithm requires to link facts extracted from documents to modifications of the ontology. We have developed a decision process that links the requirements in terms of fact extraction with operations to be performed on the ontology and the lexicon. The operations to the ontology are analyzed by the algorithm and a decision is made on the operations to be applied to the ontology. We have covered different aspects of IE either with standard approaches and, if required, we have contributed in this field. The Biomedical Domain has given attention to entities like proteins but does not cover entirely the Biomedical Domain, we have studied the annotation of different entity types where our main contribution has been on the annotation of diseases and discovered that simple methods offer a competitive performance compared to more complex methods used in the identification of genes and proteins. This means as well that disease terminology is more standardized. We have developed a system to identify relations between entities that combines co-occurrences and the classification of sentences. We identify relations that have been expressed several times in the collection. We rely on the redundancy of information in the documents. A relation between entities may be hypothetical. Unfortunately, this procedure may avoid specific knowledge that may be interesting to know, on the other hand, as we have shown, specific knowledge will not provide good results in this retrieval task.

The refinement algorithm, again, has performed different refinements on the set comprised of the ontology and the lexicon. The first one is to clean up the lexicon. The lexical entries are collected from different databases and may contain redundant terms or less specific terms than required for the retrieval task. The lexicon cleaning has proved to be effective. It has shown to target specific terms that better denote the concept without ambiguity since lexical entries in a lexicon may denote senses that are not completely disjoint. We have found as well that many terms in these lexicons never appear as such in the documents. The best technique relies on a small set of relevant documents; the increment is quite significant meaning that the terminology is rather ambiguous.

Then we have analyzed the documents to either add new terms to the lexicon and relate these terms to existing concept or to create a new concept and new relations between the concepts. Different strategies are applied to extract terms from the documents either based on the syntax of the sentence or based on named entity recognition techniques. We have seen that the strategies based on named entity recognition have a better performance since a better normalization

of the concepts is done. Selection of terms based on relevance has proved to produce better results while co-occurrences offer a similar performance and do not require any relevance information denoted explicitly. On the other hand, pseudo-relevance feedback has offered poor performance on the dataset. This behavior has already been found in previous Genomics TREC competitions where the different expansion methods did not improve over simple retrieval techniques. This may be due to the low precision at top-n documents which makes the selection of terms to drift the intention of the query. This behavior has been found as well with the relevance models, which is the baseline reformulation applied.

The refinement of terms for the relations produces interesting results if we have enough examples to build a model or prioritize the features. As we have seen for the PPI-dataset we are able to identify terms that are denoting the relations that are effective for retrieval and some of the terms are clear indicators of protein interactions or have been identified in similar works.

IE is applied on a set of documents. We have contributed to the IE pipelines from the system used and have proposed a way to obtain relevant relations from the associations expressed as co-occurrences. Therefore, almost all the proposals for refinements are provided by relations. This may mean that the method is focused on information that has enough support from the documents and the ontology already contains concepts and synsets to cover the needs of the retrieval model.

Even though we have identified differences between the knowledge required for each one of the queries, it is relevant to mention that there are several commonalities; the queries for the PGN-disease data set seem to consider interacting proteins with the query PGN. Meanwhile, there are some queries where related diseases in the hierarchy are relevant but not always. Sometimes the hyponym or the hypernym is relevant. This allows us to derive an interesting conclusion. The knowledge requirements to answer a given query are dependent on how the knowledge is expressed in the documents for a given query as we already commented above. This means that the knowledge learned is relevant for the query but it lacks part of the possible existing relations. We propose an artifact that links the ontology and the retrieval task, which for a given concept provides information about the relevance for the retrieval in a concept set.

The PPI dataset has been built out of the information available in a dataset. We have shown that there data is sufficient to derive interesting conclusions and the fact that there are potentially missing documents in this set has allowed us to test our algorithms and compare the results. The usage of noisy dataset produced from data existing in databases has already been successfully used in the literature[36] and has been successfully used in our work too.

In the following section we further analyze these conclusions to provide guidelines for future work related to this research.

## 8.2 Future Work

This work has presented an approach to refine an ontology for IR. This approach depends on several techniques provided from a variety of domains. Each of these techniques allows for different configurations with potential benefits in retrieval performance. In this section we propose specific ideas for future work.

The implementation of the ontology query model used in this work has as main purpose to show the behavior of the refinement algorithms. Different implementations of the estimation of probabilities according to specific characteristics of the dataset or the document collection are open to be considered. One of the issues identified in Medline is the different lengths between the documents and a specific smoothing prepared to consider the different document lengths will provide a better performance. In our approach we have used a unigram language model. The usage of bigrams or higher order n-grams may provide more specificity to the query terms in the model. We have to be cautious since bigram models have only been able to offer a small improvement. The combination of our approach with other existing retrieval models might be explored in the future.

We have used terms related to the topic of the query. Further research in topic detection[14] and the combination with an appropriate language model that considers the usage of negative topics[161] will further discard negative documents from the set of retrieved documents. The discovery of existing topics might help to refine the set of topics in the ontology.

In our work we have selected an estimation of the probabilities in the query model that does not consider the example documents. The probabilities could be better refined based on the provided examples so we include the probability of a term as being used to refer to a specific concept. Therefore, the lack of knowledge may bias the probability estimation to the known data. The study of the combination of the probability estimation and knowledge discovery could be integrated in an approach similar to the expectation-maximization algorithm applied to this problem, which may provide a less biased set with the expected knowledge. In addition, we have developed our work as part of the query model, but part of this estimation is translated into a refined document model. Further research is proposed based on the usage of named entity recognition techniques to handle the indexing combined with a word index or as a translation model.

Our IE system is based on existing resources and techniques and on methods that we have developed. More sophisticated term extraction will enable to find more candidate terms. Several syntactic problems hinder the system from identifying specific terms like coordination. Some work has been developed in the computational linguistic field[77] and may be applied in the recognition and resolution of entities. The types of relations that we have used cover a small part of all the possible relations and the discovery of new types of relations may identify the knowledge that we were not able to identify that are useful for retrieval. Further preparation of annotated corpora or systems available will increase the possibility of integrating these components in our system. The information extraction might profit from ontology based unsupervised sense

disambiguation techniques [11, 110] that might be further explored. In addition, the implementation of our extraction system is quite flexible and new approaches can be integrated implementing a Java interface.

The techniques proposed in this work might be of interest to other domains, even though it might be limited by the ontological resources and the document collections available. The refinement algorithm can be adapted to other tasks that are not related to information retrieval.

We have worked with Medline text but as we mentioned in Chapter 7, a Medline citation contains metadata like the MeSH annotation which may improve retrieval performance.

As more full text documents are available in Open Access Journals, a wider retrieval possibility will be available. This opens the possibility to methods like the relevance models which suffer from the size of the Medline citation. This has been found in current TREC competitions based on excerpts from full text documents, but again, the number of available documents was rather small (approx. 150K) compared to Medline (18M).

Finally, we have used synthetic sets based on relevant queries for biologists that should represent an average user but may still be refined for specific purposes. Based on these requirements, a post-processing of the retrieved documents can be proposed if more specific requirements are needed like documents dealing with therapeutic techniques; text categorization techniques may be considered. Furthermore, the size of the collections in this thesis is limited and more interesting results may be obtained if larger data sets are prepared.

## 8.3 Publications

- Chapter 2. In [83] we have presented an approach to do the link between lexicons and ontologies and have proposed the creation of a shared lexicon for the Life Sciences.
- Chapter 3. In [116] we have discussed query reformulation in the Biomedical Domain. The Ontology Query Model has been presented in [80].
- Chapter 4. In [79] we have presented the ontology refinement algorithm.
- Chapter 5. In [84] we have compared different approaches to perform disease annotation based on the UMLS Metathesaurus and in [82][78] we have used co-occurrence analysis to identify relations in the Biomedical literature.
- Chapter 7. In [80] we have presented and evaluated the Ontology Query Model and used it for the cleansing of the lexicon of BOIR. In [79] we have evaluated the ontology refinement approach. In [81] we have presented an approach to learn a model and terms related to the relation encoded in the topic template queries.

# Appendix A

# Queries

| Query Id | Query Text |
|---|---|
| 92 | Ribosomal Protein L11 and cancer |
| 93 | DRD4 and alcoholism |
| 96 | HMG and HMGB1 and hepatitis |
| 110 | Interferon-beta and Multiple Sclerosis |
| 111 | PRNP and Mad Cow Disease |
| 112 | IDE gene and Alzheimer's Disease |
| 113 | MMS2 and Cancer |
| 114 | APC (adenomatous polyposis coli) and Colon Cancer |
| 115 | Nurr-77 Parkinson's and Disease |
| 116 | insulin receptor and cancer |
| 117 | Aapolipoprotein E (ApoE) and Alzheimer's Disease |
| 118 | Transforming growth factor-beta1 (TGF-beta1) and Cerebral AmyloidAngiopathy (CAA) |
| 119 | GSTM1 and Breast Cancer |
| 130 | BRCA1 regulation of ubiquitin and cancer |
| 132 | APC (adenomatous polyposis coli) and wnt and colon cancer |
| 134 | CFTR and Sec61 degradation of CFTR and cystic fibrosis |
| 140 | BRCA1 185delAG mutation role in ovarian cancer |
| 141 | Huntingtin mutations role in Huntington's Disease |
| 145 | Mutations of hypocretin receptor 2 and narcolepsy |
| 146 | Mutations of presenilin 1 and Alzheimer's disease |

Table A.1: Protein-Disease queries

| Query Id | Query Text |
|---|---|
| 287 | DNA repair and recombination protein RAD52 AND DNA repair protein RAD51 |
| 288 | Transcriptional adapter 2 AND Histone acetyltransferase GCN5 |

| | |
|---|---|
| 289 | Heat shock protein SSC1, mitochondrial precursor AND GrpE protein homolog, mitochondrial precursor |
| 290 | Serine/threonine-protein kinase RAD53 AND DNA repair protein RAD9 |
| 291 | Protein transport protein SEC23 AND Protein transport protein Sec24 |
| 292 | Regulatory protein SIR4 AND DNA-binding protein RAP1 |
| 293 | FIP1 protein AND Poly(A) polymerase |
| 294 | Bud emergence protein 1 AND Cell division control protein 24 |
| 295 | Mitogen-activated protein kinase FUS3 AND STE5 protein |
| 297 | Helicase SGS1 AND DNA topoisomerase III |
| 298 | Ribonucleotide reductase inhibitor protein SML1 AND Ribonucleoside-diphosphate reductase large chain 1 |
| 299 | MUTL protein homolog 1 AND DNA mismatch repair protein PMS1 |
| 300 | G2/mitotic-specific cyclin 2 AND Cell division control protein 28 |
| 301 | Mitochondrial protein import protein MAS5 AND Heat shock protein SSA1 |
| 302 | Regulatory protein SIR3 AND DNA-binding protein RAP1 |
| 303 | Mitochondrial import inner membrane translocase subunit TIM17 AND Mitochondrial import inner membrane translocase subunit TIM23 |
| 304 | TATA-box binding protein AND Transcription initiation factor TFIID subunit 1 |
| 305 | Double-strand break repair protein MRE11 AND DNA repair protein RAD50 |
| 310 | Cofilin AND Actin |
| 311 | Cell division control protein 42 AND GTPase-interacting component 1 |
| 312 | Actin AND Fimbrin |
| 313 | Serine/threonine-protein kinase STE7 AND STE5 protein |
| 314 | STE5 protein AND Serine/threonine-protein kinase STE11 |
| 315 | Eukaryotic initiation factor 4F subunit p130 AND Polyadenylate-binding protein, cytoplasmic and nuclear |
| 316 | Eukaryotic initiation factor 4F subunit p150 AND Polyadenylate-binding protein, cytoplasmic and nuclear |
| 317 | Eukaryotic initiation factor 4F subunit p150 AND Eukaryotic translation initiation factor 4E |
| 318 | Guanine nucleotide-binding protein alpha-1 subunit AND Guanine nucleotide-binding protein beta subunit |
| 319 | Glucose repression regulatory protein TUP1 AND Glucose repression mediator protein |
| 320 | Carbon catabolite derepressing protein kinase AND SIP4 protein |
| 321 | AFR1 protein AND Cell division control protein 12 |
| 322 | Securin AND Separin |

| | |
|---|---|
| 323 | DNA polymerase epsilon, catalytic subunit A AND DNA primase large subunit |
| 324 | DNA polymerase epsilon, catalytic subunit A AND DNA primase small subunit |
| 325 | DNA polymerase alpha catalytic subunit AND DNA polymerase epsilon, catalytic subunit A |
| 326 | Eukaryotic translation initiation factor 3 90 kDa subunit AND Eukaryotic translation initiation factor 3 39 kDa subunit |
| 327 | Spindle pole body component SPC97 AND Spindle pole body component SPC98 |
| 328 | DNA mismatch repair protein MSH2 AND MUTS protein homolog 6 |
| 329 | Mitosis inhibitor protein kinase SWE1 AND Protein arginine N-methyltransferase HSL7 |
| 330 | Nucleoporin NUP116/NSP116 AND Nucleoporin GLE2 |
| 331 | Serine/threonine-protein kinase RAD53 AND Anti-silencing protein 1 |
| 332 | Mitochondrial import receptor subunit TOM22 AND Mitochondrial import receptor subunit TOM20 |
| 333 | DNA topoisomerase II AND DNA topoisomerase II |
| 334 | mRNA capping enzyme alpha subunit AND mRNA capping enzyme beta subunit |
| 335 | ATP-dependent molecular chaperone HSP82 AND Peptidyl-prolyl cis-trans isomerase CYP7 |
| 336 | Protein transport protein SEC23 AND SED5-binding protein 2 |
| 337 | Cell division control protein 4 AND Cell division control protein 6 |
| 338 | TATA-box binding protein AND Transcription initiation factor IIA small chain |
| 339 | TATA-box binding protein AND Transcription initiation factor IIA large chain |
| 340 | RNA polymerase I specific transcription initiation factor RRN7 AND RNA polymerase I specific transcription initiation factor RRN6 |
| 341 | DNA repair protein RAD10 AND DNA repair protein RAD1 |
| 342 | Eukaryotic translation initiation factor 3 93 kDa subunit AND Eukaryotic translation initiation factor 5 |
| 343 | PAN1 protein AND END3 protein |
| 344 | Double-strand break repair protein MRE11 AND DNA repair protein XRS2 |
| 345 | Double-strand break repair protein MRE11 AND Double-strand break repair protein MRE11 |
| 346 | URE2 protein AND Nitrogen regulatory protein GLN3 |
| 347 | Cell division control protein 7 AND DBF4 protein |
| 348 | Eukaryotic initiation factor 4F subunit p20 AND Eukaryotic translation initiation factor 4E |

| | |
|---|---|
| 349 | Mitochondrial regulator of splicing 5 AND Mitochondrial import inner membrane translocase subunit TIM22 |
| 350 | DNA repair protein RAD51 AND DNA repair and recombination protein RAD54 |
| 352 | RHO1 protein AND Protein kinase C-like 1 |
| 353 | Cell division control protein 42 AND Serine/threonine-protein kinase CLA4 |
| 354 | Cell division control protein 42 AND Cell division control protein 24 |
| 355 | Adenylyl cyclase-associated protein AND Adenylate cyclase |
| 356 | Adenylyl cyclase-associated protein AND Actin binding protein |
| 357 | STE50 protein AND Serine/threonine-protein kinase STE11 |
| 358 | Ribonucleoside-diphosphate reductase large chain 1 AND Ribonucleoside-diphosphate reductase large chain 1 |
| 359 | DNA repair protein RAD16 AND DNA repair protein RAD7 |
| 360 | Paired amphipathic helix protein SIN3 AND Histone deacetylase RPD3 |
| 361 | CRE-binding bZIP protein SKO1 AND Mitogen-activated protein kinase HOG1 |
| 362 | INO4 protein AND INO2 protein |
| 363 | Importin alpha subunit AND Nucleoporin NUP2 |
| 364 | Nucleoporin NUP1 AND Importin beta-1 subunit |
| 365 | Ribonucleoside-diphosphate reductase small chain 2 AND Ribonucleoside-diphosphate reductase small chain 1 |
| 366 | GTP-binding nuclear protein GSP1/CNR1 AND Importin beta-1 subunit |
| 367 | GTP-binding nuclear protein GSP1/CNR1 AND Importin alpha re-exporter |
| 368 | GTP-binding nuclear protein GSP1/CNR1 AND Ran-specific GTPase-activating protein 1 |
| 369 | DNA-directed RNA polymerase II 32 kDa polypeptide AND DNA-directed RNA polymerase II 19 kDa polypeptide |
| 370 | Regulatory protein PHO2 AND Phosphate system positive regulatory protein PHO4 |
| 371 | Regulatory protein PHO2 AND Myb-like DNA-binding protein BAS1 |
| 372 | SAC2 protein AND Hypothetical 95.4 kDa protein in MAD2-RNR2 intergenic region |
| 373 | Glucose repression regulatory protein TUP1 AND Glucose repression regulatory protein TUP1 |
| 374 | Glucose repression regulatory protein TUP1 AND CRE-binding bZIP protein SKO1 |
| 375 | Carbon catabolite derepressing protein kinase AND SIP2 protein |
| 376 | Carbon catabolite derepressing protein kinase AND Nuclear protein SNF4 |
| 377 | Carbon catabolite derepressing protein kinase AND SIP1 protein |

| | |
|---|---|
| 378 | ATP11 protein, mitochondrial precursor AND ATP synthase beta chain, mitochondrial precursor |
| 379 | Cell division control protein 3 AND Probable serine/threonine-protein kinase YKL101W |
| 380 | Structural maintenance of chromosome 1 AND Cohesin subunit SCC3 |
| 381 | Pre-mRNA splicing factor PRP9 AND Pre-mRNA splicing factor PRP21 |
| 382 | Pre-mRNA splicing factor PRP21 AND Pre-mRNA splicing factor PRP11 |
| 383 | Heat shock protein SSC1, mitochondrial precursor AND Import inner membrane translocase subunit TIM44, mitochondrial precursor |
| 384 | Protein arginine N-methyltransferase HSL7 AND Probable serine/threonine-protein kinase YKL101W |
| 385 | Mitochondrial import receptor subunit TOM20 AND Mitochondrial import receptor subunit TOM40 |
| 386 | 60S ribosomal protein L10 AND Ribosome assembly protein SQT1 |
| 387 | Retrograde regulation protein 1 AND Retrograde regulation protein 3 |
| 388 | Integral membrane protein SED5 AND SLY1 protein |
| 389 | Integral membrane protein SED5 AND Hypothetical 25.4 kDa protein in GUT1-RIM1 intergenic region |
| 390 | Integral membrane protein SED5 AND SFT1 protein |
| 391 | Phosphate system cyclin PHO80 AND Phosphate system positive regulatory protein PHO81 |
| 392 | Protein transport protein SEC61 gamma subunit AND Protein transport protein SEC61 alpha subunit |
| 393 | DNA repair helicase RAD3 AND Suppressor of stem-loop protein 1 |
| 394 | Tubulin gamma chain AND Spindle pole body component SPC97 |
| 395 | Tubulin gamma chain AND Spindle pole body component SPC98 |
| 396 | Proliferating cell nuclear antigen AND DNA polymerase delta subunit 3 |
| 397 | DNA mismatch repair protein MSH2 AND MUTS protein homolog 3 |
| 398 | DNA mismatch repair protein MSH2 AND Exodeoxyribonuclease I |
| 399 | Transcription factor SKN7 AND Transcription factor SKN7 |
| 400 | 78 kDa glucose-regulated protein homolog precursor AND Translocation protein |
| 401 | Transcription initiation factor IIA large chain AND Transcription initiation factor IIA small chain |
| 402 | Translational activator GCN1 AND GCN20 protein |
| 403 | Regulatory protein SWI4 AND Mitogen-activated protein kinase SLT2/MPK1 |

| | |
|---|---|
| 404 | Cell cycle protein kinase DBF2 AND Maintenance of ploidy protein MOB1 |
| 405 | Eukaryotic translation initiation factor 5 AND Eukaryotic translation initiation factor 2 beta subunit |
| 406 | Nonsense-mediated mRNA decay protein 2 AND NAM7 protein |
| 407 | Vacuolar ATP synthase subunit B AND Vacuolar ATP synthase catalytic subunit A |
| 408 | DBF4 protein AND Serine/threonine-protein kinase RAD53 |
| 409 | Origin recognition complex subunit 1 AND Regulatory protein SIR1 |
| 410 | Suppressor protein MPT5 AND SST2 protein |
| 411 | NAD-dependent histone deacetylase SIR2 AND Transcription regulatory protein SNF12 |
| 412 | ATP-dependent molecular chaperone HSP82 AND Heat shock protein STI1 |
| 413 | Cell division control protein 25 AND Ras-like protein 2 |
| 414 | Cell division control protein 25 AND ATP-dependent molecular chaperone HSP82 |
| 415 | Serine/threonine-protein kinase GIN4 AND Cell division control protein 3 |
| 416 | Transcriptional activator of sulfur metabolism MET4 AND Transcriptional activator of sulfur metabolism MET28 |
| 417 | Centromere-binding protein 1 AND Transcriptional activator of sulfur metabolism MET4 |
| 418 | Protein transport protein SEC23 AND SED5-binding protein 3 |
| 419 | Protein transport protein SEC23 AND Multidomain vesicle coat protein |
| 420 | Cyclin-dependent kinase inhibitor FAR1 AND G1/S-specific cyclin CLN2 |
| 421 | Cell division control protein 24 AND Ras-related protein RSR1 |
| 422 | Cell division control protein 24 AND Cyclin-dependent kinase inhibitor FAR1 |
| 423 | Regulatory protein SIR4 AND NAD-dependent histone deacetylase SIR2 |
| 424 | Histone H4 AND Regulatory protein SIR3 |
| 425 | Vesicle transport v-SNARE protein VTI1 AND Vesicular-fusion protein SEC17 |
| 426 | Vesicle transport v-SNARE protein VTI1 AND Integral membrane protein SED5 |
| 427 | Hypothetical 47.4 kDa protein in OPY1-AGP2 intergenic region AND Myosin-4 isoform |
| 428 | Vesicular transport protein BOS1 AND Protein transport protein BET1 |
| 429 | GTP-binding protein YPT1 AND Rab proteins geranylgeranyltransferase component A |

| | |
|---|---|
| 430 | RNA polymerase II holoenzyme cyclin-like subunit AND Meiotic mRNA stability protein kinase UME5 |
| 431 | Protein transport protein TIP20 AND Hypothetical 88.1 kDa protein in ATX1-SIP3 intergenic region |
| 432 | IKI1 protein AND Hypothetical 89.4 kDa Trp-Asp repeats containing protein in PMT6-PCT1 intergenic region |
| 433 | POP2 protein AND Glucose-repressible alcohol dehydrogenase transcriptional effector |
| 434 | Protein transport protein SEC13 AND WEB1 protein |
| 435 | DNA damage response protein kinase DUN1 AND Ribonucleotide reductase inhibitor protein SML1 |
| 436 | Peroxisomal membrane protein PEX14 AND Peroxisomal membrane protein PAS20 |
| 437 | Syntaxin VAM3 AND Vacuolar protein sorting 33 |
| 438 | Vacuolar morphogenesis protein VAM7 AND Syntaxin VAM3 |
| 439 | Transcriptional regulator UME6 AND Meiosis-inducing protein 1 |
| 440 | Transcriptional regulator UME6 AND Serine/threonine-protein kinase MDS1/RIM11 |
| 441 | TEM1 protein AND Cell division control protein 15 |
| 442 | Ubiquitin ligase complex F-box protein GRR1 AND G1/S-specific cyclin CLN2 |
| 443 | Ubiquitin-conjugating enzyme E2-34 kDa AND Cell division control protein 4 |
| 444 | TATA-box binding protein AND SPT3 protein |
| 445 | TATA-box binding protein AND Transcription factor IIIB 70 kDa subunit |
| 446 | TATA-box binding protein AND Transcriptional adapter 2 |
| 447 | TATA-box binding protein AND Importin beta-5 subunit |
| 448 | Mitogen-activated protein kinase HOG1 AND Tyrosine-protein phosphatase 2 |
| 449 | Ubiquitin-conjugating enzyme E2-20 kDa AND N-end-recognizing protein |
| 450 | Ubiquitin AND Transcriptional activator of sulfur metabolism MET4 |
| 451 | Peroxisomal targeting signal 2 receptor AND Peroxisomal targeting signal receptor |
| 452 | Peroxisomal targeting signal 2 receptor AND Hypothetical 32.0 kDa protein in REC104-SOL3 intergenic region |
| 453 | Negative regulator of the PHO system AND Phosphate system positive regulatory protein PHO81 |
| 454 | Protein transport protein SEC9 AND Synaptobrevin homolog 1 |
| 455 | Protein transport protein SEC9 AND SSO1 protein |
| 456 | MET30 protein AND Transcriptional activator of sulfur metabolism MET4 |
| 457 | Checkpoint protein MEC3 AND DNA damage checkpoint control protein RAD17 |

| | |
|---|---|
| 458 | Nucleosome assembly protein AND NAP1-binding protein |
| 459 | Nucleosome assembly protein AND Serine/threonine-protein kinase GIN4 |
| 460 | Nuclear polyadenylated RNA-binding protein 4 AND mRNA 3'-end processing protein RNA15 |
| 461 | Transcriptional activator HAP2 AND Transcriptional activator HAP5 |
| 463 | Serine/threonine protein phosphatase PP1-2 AND Protein phosphatase 1 regulatory subunit GAC1 |
| 464 | Translation initiation factor eIF-2B gamma subunit AND Translation initiation factor eIF-2B delta subunit |
| 465 | Translation initiation factor eIF-2B gamma subunit AND Translation initiation factor eIF-2B epsilon subunit |
| 466 | Translation initiation factor eIF-2B gamma subunit AND Translation initiation factor eIF-2B alpha subunit |
| 467 | Acetolactate synthase small subunit, mitochondrial precursor AND Acetolactate synthase, mitochondrial precursor |
| 468 | Kinetochore assembly protein DAM1 AND Chromosome partition protein DUO1 |
| 469 | Dihydrolipoyl dehydrogenase, mitochondrial precursor AND Dihydrolipoyllysine-residue succinyltransferase component of 2-oxoglutarate dehydrogenase complex, mitochondrial precursor |
| 470 | Cyclophilin seven suppressor 1 AND Peptidyl-prolyl cis-trans isomerase CYP7 |
| 471 | FK506-binding protein 1 AND Phosphatidylinositol 3-kinase TOR2 |
| 474 | Cell division control protein 7 AND DNA replication licensing factor MCM2 |
| 475 | Cell division control protein 53 AND Cell division control protein 4 |
| 476 | Cell division control protein 53 AND Ubiquitin-conjugating enzyme E2-34 kDa |
| 477 | Cell division control protein 13 AND Telomere elongation protein |
| 478 | Eukaryotic translation initiation factor 4E AND Polyadenylate-binding protein, cytoplasmic and nuclear |
| 479 | Autophagy protein APG7 AND Autophagy protein 8 [Contains: Apg8FG] |
| 480 | Conserved oligomeric Golgi complex component 2 AND Conserved oligomeric Golgi complex component 3 |
| 483 | Translation initiation factor eIF-2B beta subunit AND Translation initiation factor eIF-2B gamma subunit |
| 484 | SCO1 protein, mitochondrial precursor AND SCO1 protein, mitochondrial precursor |
| 485 | Cytochrome c oxidase polypeptide II precursor AND SCO1 protein, mitochondrial precursor |

| | |
|---|---|
| 486 | Postreplication repair protein RAD18 AND Ubiquitin-conjugating enzyme E2-20 kDa |
| 487 | Mitochondrial import inner membrane translocase subunit TIM10 AND Mitochondrial import inner membrane translocase subunit TIM9 |
| 488 | Pre-mRNA splicing factor PRP19 AND Pre-mRNA splicing factor PRP19 |
| 489 | DNA-directed RNA polymerase II largest subunit AND Transcription elongation factor S-II |
| 490 | DNA-directed RNA polymerase II largest subunit AND mRNA capping enzyme beta subunit |
| 491 | DNA-directed RNA polymerase II largest subunit AND mRNA capping enzyme alpha subunit |
| 492 | DNA-directed RNA polymerase II largest subunit AND PCF11 protein |
| 493 | DNA repair protein RAD51 AND DNA repair protein RAD55 |
| 494 | DNA repair protein RAD51 AND DNA repair protein RAD51 |
| 495 | Mannan polymerase complexes MNN9 subunit AND Mannan polymerase I complex VAN1 subunit |
| 496 | Mannan polymerase complexes MNN9 subunit AND Mannan polymerase II complex ANP1 subunit |
| 498 | Tubulin alpha-1 chain AND PAC2 protein |
| 499 | Transcription initiation factor TFIID subunit 10 AND Transcription initiation factor TFIID subunit 14 |
| 500 | Galactose/lactose metabolism regulatory protein GAL80 AND GAL3 protein |
| 502 | Regulatory protein GAL4 AND 26S protease regulatory subunit 8 homolog |
| 503 | Transcriptional adapter 2 AND Transcriptional adapter 3 |
| 504 | RHO1 protein AND Rho-GTPase-activating protein LRG1 |
| 505 | RHO1 protein AND Bud emergence protein 4 |
| 506 | RHO1 protein AND Exocyst complex component SEC3 |
| 507 | RHO1 protein AND 1,3-beta-glucan synthase component GLS1 |
| 508 | RHO1 protein AND RHO1 GDP-GTP exchange protein 2 |
| 509 | Proline-rich protein LAS17 AND Actin-like protein ARP2 |
| 510 | Cell division control protein 42 AND Bud emergence protein 4 |
| 511 | Bud emergence protein 1 AND Cyclin-dependent kinase inhibitor FAR1 |
| 512 | Bud emergence protein 1 AND BOI2 protein |
| 513 | Bud emergence protein 1 AND BOB1 protein |
| 514 | Bud emergence protein 1 AND STE5 protein |
| 515 | Profilin AND Actin |
| 516 | Actin AND Verprolin |
| 517 | Adenylyl cyclase-associated protein AND Actin |
| 518 | Actin binding protein AND Cytoskeleton assembly control protein SLA1 |

| | |
|---|---|
| 519 | Actin binding protein AND Fimbrin |
| 520 | Actin binding protein AND Reduced viability upon starvation protein 167 |
| 521 | Import inner membrane translocase subunit TIM44, mitochondrial precursor AND GrpE protein homolog, mitochondrial precursor |
| 522 | Mating-type protein ALPHA2 AND Glucose repression regulatory protein TUP1 |
| 523 | Mating-type protein ALPHA2 AND Mating-type protein A1 |
| 524 | Ras-like protein 1 AND Cell division control protein 25 |
| 525 | Mitogen-activated protein kinase KSS1 AND STE5 protein |
| 526 | Serine/threonine-protein kinase STE11 AND Polymyxin B resistance protein kinase |
| 527 | STE5 protein AND Guanine nucleotide-binding protein beta subunit |
| 528 | Cell division control protein 28 AND G1/S-specific cyclin CLN2 |
| 529 | Cell division control protein 28 AND Cell division control protein 6 |
| 530 | Cell division control protein 28 AND Serine/threonine-protein kinase CAK1 |
| 531 | Cell division control protein 28 AND G1/S-specific cyclin CLN3 |
| 532 | Cell division control protein 28 AND G2/mitotic-specific cyclin 3 |
| 533 | DNA repair protein RAD16 AND Helicase SGS1 |
| 534 | Pre-mRNA splicing factor SLU7 AND Pre-mRNA splicing factor PRP18 |
| 535 | PCF11 protein AND mRNA 3'-end processing protein RNA15 |
| 536 | PCF11 protein AND mRNA 3'-end processing protein RNA14 |
| 537 | CUS1 protein AND HSH49 protein |
| 538 | Calmodulin AND NUF1 protein |
| 539 | Calmodulin AND Myosin-2 |
| 540 | Calmodulin AND Calcium/calmodulin-dependent protein kinase II |
| 541 | Calmodulin AND Calcium/calmodulin-dependent protein kinase I |
| 542 | Calmodulin AND Serine/threonine protein phosphatase 2B catalytic subunit A1 |
| 543 | Ubiquitin-conjugating enzyme E2 13 AND Ubiquitin-conjugating enzyme variant MMS2 |
| 544 | Kinesin-like protein KAR3 AND Spindle pole body associated protein |
| 545 | Regulatory protein MIG1 AND Glucose repression mediator protein |
| 546 | Importin alpha subunit AND Importin alpha re-exporter |
| 547 | Nucleoporin NUP1 AND Importin alpha subunit |
| 549 | GTP-binding nuclear protein GSP1/CNR1 AND Exportin 1 |

| | |
|---|---|
| 550 | GTP-binding nuclear protein GSP1/CNR1 AND Ran-specific GTPase-activating protein 2 |
| 551 | GTP-binding nuclear protein GSP1/CNR1 AND MOG1 protein |
| 552 | Tyrosine-protein phosphatase CDC14 AND Nucleolar protein NET1 |
| 553 | Nucleoporin NIC96 AND Nucleoporin POM152 |
| 554 | Nucleoporin NUP82 AND Nucleoporin NSP1 |
| 555 | DNA replication regulator DPB11 AND DNA replication regulator SLD2 |
| 556 | Dolichyl-diphosphooligosaccharide–protein glycosyltransferase beta subunit precursor AND Dolichyl-diphosphooligosaccharide–protein glycosyltransferase delta subunit precursor |
| 557 | Dolichyl-diphosphooligosaccharide–protein glycosyltransferase beta subunit precursor AND Dolichyl-diphosphooligosaccharide–protein glycosyltransferase alpha subunit precursor |
| 558 | Guanine nucleotide-binding protein alpha-1 subunit AND Serine/threonine-protein kinase STE11 |
| 559 | Guanine nucleotide-binding protein beta subunit AND Cell division control protein 24 |
| 560 | Hypothetical 88.1 kDa protein in ATX1-SIP3 intergenic region AND Coatomer delta subunit |

Table A.2: Protein-protein interaction in yeast queries

# Appendix B

# Sample Medline Entry

```
<MedlineCitation Owner="NLM" Status="MEDLINE">
  <PMID>17107631</PMID>
  <DateCreated>
    <Year>2006</Year>
    <Month>11</Month>
    <Day>19</Day>
  </DateCreated>
  <DateCompleted>
    <Year>2007</Year>
    <Month>10</Month>
    <Day>09</Day>
  </DateCompleted>
  <Article PubModel="Electronic">
    <Journal>
      <ISSN IssnType="Electronic">1462-3994</ISSN>
      <JournalIssue CitedMedium="Internet">
        <Volume>8</Volume>
        <Issue>26</Issue>
        <PubDate>
          <Year>2006</Year>
        </PubDate>
      </JournalIssue>
      <Title>Expert reviews in molecular medicine</Title>
    </Journal>
    <ArticleTitle>Refractory juvenile idiopathic arthritis: using
    autologous stem cell transplantation as a treatment strategy.
    </ArticleTitle>
    <Pagination>
      <MedlinePgn>1-11</MedlinePgn>
    </Pagination>
    <Abstract>
```

```
    <AbstractText>Cellular immune therapy for severe autoimmune
    diseases can now be considered when such patients are
    refractory to conventional treatment. The use of autologous
    stem cell transplantation (ASCT) to treat human autoimmune
    diseases has been initiated following promising results in
    a variety of animal models. Anecdotal observations have been
    made of autoimmune disease remission in patients who have
    undergone allogeneic bone marrow transplantation as a result
    of coincidental haematological malignancies. The possibility
    of inducing immunological self-tolerance by ASCT is
    particularly attractive as a means for treating juvenile
    idiopathic arthritis (JIA). In this disease, ASCT restores
    self-tolerance both through a cell-intrinsic mechanism,
    involving the reprogramming of autoreactive T cells, and
    through a cell-extrinsic mechanism, involving a renewal of
    the immune balance between CD4+CD25+ regulatory T cells
    and other T cells. This review describes the clinical results
    of ASCT performed for this disease and the possible underlying
    immunological mechanisms.</AbstractText>
  </Abstract>
  <Affiliation>Department of Pediatric Immunology, University
  Medical Center Utrecht, Wilhelmina Children's Hospital, 3508
  AB Utrecht, The Netherlands. n.wulffraat@umcutrecht.nl
  </Affiliation>
  <AuthorList CompleteYN="Y">
    <Author ValidYN="Y">
      <LastName>Wulffraat</LastName>
      <ForeName>Nico M</ForeName>
      <Initials>NM</Initials>
    </Author>
    <Author ValidYN="Y">
      <LastName>de Kleer</LastName>
      <ForeName>Ism M</ForeName>
      <Initials>IM</Initials>
    </Author>
    <Author ValidYN="Y">
      <LastName>Prakken</LastName>
      <ForeName>Berent</ForeName>
      <Initials>B</Initials>
    </Author>
  </AuthorList>
  <Language>eng</Language>
  <PublicationTypeList>
    <PublicationType>Journal Article</PublicationType>
    <PublicationType>Research Support, Non-U.S. Gov't
    </PublicationType>
```

```
      <PublicationType>Review</PublicationType>
    </PublicationTypeList>
    <ArticleDate DateType="Electronic">
      <Year>2006</Year>
      <Month>11</Month>
      <Day>15</Day>
    </ArticleDate>
  </Article>
  <MedlineJournalInfo>
    <Country>England</Country>
    <MedlineTA>Expert Rev Mol Med</MedlineTA>
    <NlmUniqueID>100939725</NlmUniqueID>
  </MedlineJournalInfo>
  <CitationSubset>IM</CitationSubset>
  <MeshHeadingList>
    <MeshHeading>
      <DescriptorName MajorTopicYN="N">Animals</DescriptorName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName MajorTopicYN="N">Arthritis, Juvenile
      Rheumatoid</DescriptorName>
      <QualifierName MajorTopicYN="N">immunology</QualifierName>
      <QualifierName MajorTopicYN="Y">surgery</QualifierName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName MajorTopicYN="N">Humans</DescriptorName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName MajorTopicYN="N">Remission,
      Spontaneous</DescriptorName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName MajorTopicYN="Y">Stem Cell
      Transplantation</DescriptorName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName MajorTopicYN="N">Transplantation,
      Autologous</DescriptorName>
    </MeshHeading>
    <MeshHeading>
      <DescriptorName MajorTopicYN="N">Treatment Outcome
      </DescriptorName>
    </MeshHeading>
  </MeshHeadingList>
  <NumberOfReferences>58</NumberOfReferences>
</MedlineCitation>
```

# Bibliography


[1] E. Agichtein and L. Gravano. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the Fifth ACM International Conference on Digital Libraries*, 2000.

[2] E. Agirre, O. Ansa, E. Hovy, and D. Martinez. Enriching very large ontologies using the www. In *Proceedings of the Ontology Learning Workshop, ECAI, Berlin, Germany*, 2000.

[3] E. Agirre and G. Rigau. Word sense disambiguation using conceptual density. In *Proceedings of the 16th conference on Computational linguistics, August*, pages 05–09, 1996.

[4] E. Alfonseca and S. Manandhar. Improving an ontology refinement method with hyponymy patterns. In *Language Resources and Evaluation (LREC-2002)*, Las Palmas, 2002.

[5] E. Alfonseca and S. Manandhar. An unsupervised method for general named entity recognition and automated concept discovery. In *Proceedings of the 1st International Conference on General WordNet*, 2002.

[6] G. Amati and P. Bruza. A logical approach to query reformulation motivated from belief change. In *Workshop on logical and uncertainty models for Information Systems*. University College London (UCL), London, England, 1999.

[7] S. Ananiadou and G. Nenadic. Automatic terminology management in biomedicine. *Text mining for biology and biomedicine. Artech House*, pages 67–97, 2006.

[8] A. R. Aronson. Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. Technical report, 2001.

[9] A. R. Aronson and T. C. Rindflesch. Query expansion using the UMLS Metathesaurus. In *Amia*, 1997.

[10] J. Bai, D. Song, P. Bruza, J.Y. 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.

[11] S. Banerjee and T. Pedersen. An adapted Lesk algorithm for word sense disambiguation using WordNet. *Lecture notes in computer science*, pages 136–145, 2002.

[12] E. Beisswanger, M. Poprat, and U. Hahn. Lexical Properties of OBO Ontology Class Names and Synonyms. In *3rd International Symposium on Semantic Mining in Biomedicine*, 2008.

[13] 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.

[14] R. Berlanga-Llavori, H. Anaya-Snchez, A. Pons-Porrata, and E. Jimnez-Ruiz. Conceptual subtopic identification in the medical domain. In Hector Geffner, Rui Prada, Isabel Machado Alexandre, and Nuno David, editors, *IBERAMIA*, volume 5290 of *Lecture Notes in Computer Science*, pages 312–321. Springer, 2008.

[15] C. Blaschke, L. Hirschman, and A. Valencia. Information extraction in molecular biology. *Briefings in Bioinformatics*, 3(2):154–165, 2002.

[16] C. Blaschke and A. Valencia. Automatic Ontology Construction from the Literature. *Genome Informatics Series*, pages 201–213, 2002.

[17] O. Bodenreider. Lexical, terminological and ontological resources for biological text mining. In *Text mining for biology and biomedicine. Artech House*, 2006.

[18] R. C. Bodner and F. Song. Knowledge-based approaches to query expansion in information retrieval. In *Canadian Conference on AI*, pages 146–158, 1996.

[19] J. Brank, M. Grobelnik, and D. Mladeni. A survey of ontology evaluation techniques. In *Proceedings of SIKDD*, 2005.

[20] 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.

[21] C. Buckley, G. Salton, J. Allan, and A. Singhal. Automatic query expansion using SMART: TREC 3. In *Text REtrieval Conference*, 1994.

[22] R. Bunescu, R. Ge, R. Kate, E. Marcotte, R. Mooney, A. Ramani, and Y. Wong. Comparative experiments on learning information extractors for proteins and their interactions. In *Comparative Experiments on Learning Information Extractors for Proteins and their Interactions*. Journal of Artificial Intelligence in Medicine, 2004.

[23] M. E. Califf and R. J. Mooney. Relational learning of pattern-match rules for information extraction. In *AAAI/IAAI*, pages 328–334, 1999.

[24] M.E. Califf and R. J. Mooney. Bottom-up relational learning of pattern matching rules for information extraction. *Journal of Machine Learning Research*, 4:177–210, 2003.

[25] G. Cao, J.Y. Nie, and J. Bai. Integrating word relationships into language models. *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 298–305, 2005.

[26] S.A. Caraballo. Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 120–126. Association for Computational Linguistics Morristown, NJ, USA, 1999.

[27] C. Cardie. Empirical methods in information extraction. *AI Magazine*, 18(4):65–80, 1997.

[28] L. Chen, H. Liu, and C. Friedman. Gene name ambiguity of eukaryotic nomenclatures. *Bioinformatics*, 21(2):248–256, January 2005.

[29] W.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(CIICM), Taiwan, 5(2)*, 2002.

[30] P. Cimiano, A. Pivk, L. Schmidt-Thieme, and S. Staab. Learning taxonomic relations from heterogeneous sources of evidence. *Ontology Learning from Text: Methods, Evaluation and Applications*, pages 59–73, 2005.

[31] F. Ciravegna, A. Dingli, Y. Wilks, and D. Petrelli. Adaptive information extraction for document annotation in amilcare. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 451–451. ACM Press, 2002.

[32] C.W. Cleverdon. The Cranfield tests on index language devices. In *Aslib Proceedings*, volume 19, pages 173–193. MCB UP Ltd, 1967.

[33] K.B. Cohen, M. Palmer, and L. Hunter. Nominalization and Alternations in Biomedical Language. *PLoS ONE*, 3(9), 2008.

[34] O. Corcho and A. Gomez-Perez. Evaluating knowledge representation and reasoning capabilities of ontology specification languages. In *Proceedings of the ECAI 2000 Workshop on Applications of Ontologies and Problem-Solving Methods*, volume 3, 2000.

[35] J. Cowie and W. Lehnert. Information extraction. *Communication ACM*, 39(1):80–91, 1996.

[36] M. Craven and J. Kumlien. Constructing biological knowledge-bases by extracting information from text sources. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*, pages 77–86, Germany, 1999.

[37] M. Craven and S. Slattery. Relational learning with statistical predicate invention: Better models for hypertext. *Machine Learning*, 43(1/2):97–119, 2001.

[38] W.B. Croft and D.J. Harper. Using probabilistic models of document retrieval without relevance information. *Journal of Documentation*, 35(4):285–295, 1979.

[39] D. R. Cutting, J. O. Pedersen, D. Karger, and J. W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 318–329, 1992.

[40] I. Dagan, S. Marcus, and S. Markovitch. Contextual word similarity and estimation from sparse data. In *Proceedings of the 31st conference on Association for Computational Linguistics*, pages 164–171. Association for Computational Linguistics, 1993.

[41] P.A.C. David, F.B. Bernard, B.L. William, and T.J. David. BioRAT: extracting biological information from full-length papers. *Bioinformatics*, 20(17):3206, 2004.

[42] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[43] T.G. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10:1895–1923, 1998.

[44] A. Divoli and T.K. Attwood. BioIE: extracting informative sentences from the biomedical literature. *Bioinformatics*, 21(9):2138–2139, 2005.

[45] I. Donaldson, J. Martin, B. de Bruijn, and C. Wolting. Prebind and textomy - mining the biomedical literature for proteinprotein interactions using a support vector machine, 2003.

[46] E. Efthimiadis. Query expansion. In *Appeared in: Williams, Martha E., ed Annual Review of Information Systems and Technologies (ARIST), v31*, pages 121–187, 1996.

[47] A. Faatz and R. Steinmetz. Ontology enrichment with texts from the www. In *Semantic Web Mining, WS02*, 2002.

[48] D. Faure and C. Nédellec. A corpus-based conceptual clustering method for verb frames and ontology acquisition. In *LREC workshop on adapting lexical and corpus resources to sublanguages and applications*, pages 707–728, 1998.

[49] D. Faure and C. Nédellec. ASIUM: learning subcategorization frames and restrictions of selection. In Y. Kodratoff, editor, *10th European Conference on Machine Learning, Workshop on Text Mining*, 1998.

[50] T. Fawcett. Roc graphs: Notes and practical considerations for researchers. Technical report, HP Laboratories, MS 1143, 1501 Page Mill Road, Palo Alto, CA 94304, 2004.

[51] A. Finn and N. Limerick. Active learning selection strategies for information extraction. In *Proceedings of the ECML-2004 Workshop on Adaptive Text Extraction and Mining (ATEM-2003*, 2003.

[52] E.A. Fox. Lexical relations: Enhancing effectiveness of information retrieval systems. *SIGIR Forum*, 15(3):5–36, 1980.

[53] K. Frantzi, S. Ananiadou, and H. Mima. Automatic recognition of multiword terms:. the c-value/nc-value method. *International Journal on Digital Libraries*, V3(2):115–130, 2000.

[54] D. Freitag. Multistrategy learning for information extraction. In *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*, pages 161–169. Morgan Kaufmann Publishers Inc., 1998.

[55] K. Fukuda, T. Tsunoda, A. Tamura, and T. Takagi. Toward information extraction: Identifying protein names from biological papers. In *Proc. of the Pacific Symposium on Biocomputing*, 1998.

[56] A. Gangemi, D.M. Pisanelli, and G. Steve. Ontology Integration: Experiences with Medical Terminologies. In *Formal Ontology in Information Systems: Proceedings of the First International Conference (FOIS'98), June 6-8, Trento, Italy*. Ios Pr Inc, 1998.

[57] S. Gauch and J.B. Smith. An expert system for automatic query reformation. *Journal of the American Society of Information Science*, 44(3):124–136, 1993.

[58] S. Gaudan, A.J. Yepes, V. Lee, and D. Rebholz-Schuhmann. Combining Evidence, Specificity, and Proximity towards the Normalization of Gene Ontology Terms in Text. *EURASIP Journal on Bioinformatics and Systems Biology*, 8(1), 2008.

[59] J. Gonzalo, F. Verdejo, I. Chugur, and J. Cigarran. Indexing with WordNet synsets can improve text retrieval. In *Proceedings of the COLING/ACL*, volume 98, pages 38–44, 1998.

[60] W.R. Greiff. A theory of term weighting based on exploratory data analysis. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 11–19, New York, NY, USA, 1998. ACM.

[61] T. R. Gruber. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. In N. Guarino and R. Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*, Deventer, The Netherlands, 1993. Kluwer Academic Publishers.

[62] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[63] N. Guarino. Semantic matching: Formal ontological distinctions for information organization, extraction, and integration. In *SCIE*, pages 139–170, 1997.

[64] N. Guarino. Formal Ontology and Information Systems. In *Formal Ontology in Information Systems: Proceedings of the First International Conference (FOIS'98), June 6-8, Trento, Italy*. Ios Pr Inc, 1998.

[65] N. Guarino and R. Poli. Formal ontology in conceptual analysis and knowledge representation. *Special issue of the International Journal of Human and Computer Studies, vol. 43 n. 5/6, Academic Press.*, 1995.

[66] U. Hahn and K. Schnattinger. Towards text knowledge engineering. In *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*, pages 524–531. American Association for Artificial Intelligence, 1998.

[67] M.A. Hearst. Automatic acquisition of hyponyms from large text corpora. Technical Report S2K-92-09, 1992.

[68] M.A. Hearst. Automated discovery of wordnet relations. In MIT Press, editor, *WordNet: An Electronic Lexical Database and Some of its Applications*, 1998.

[69] M.A. Hearst, D.R. Karger, and J.O. Pedersen. Scatter/gather as a tool for the navigation of retrieval results. In *Working Notes AAAI Fall Symp. AI Applications in Knowledge Navigation*, 1995.

[70] M.A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *SIGIR*, pages 76–84, 1996.

[71] M.A. Hearst and H. Schutze. Customizing a lexicon to better suit a computational task. In *Proc. of the SIGLEX Workshop on Acquisition of Lexical Knowledge from Text, Columbus Ohio*, 1993.

[72] D. Hiemstra and A. Vries. Relating the new language models of information retrieval to the traditional retrieval models. Technical Report TR-CTIT-00-09, 2000.

[73] G. Hirst. Ontology and the lexicon. In *Handbook on Ontologies in Information Systems*, pages 209–230. Springer, 2004.

[74] W.T. Hole and S. Srinivasan. Discovering missed synonymy in a large concept-oriented metathesaurus. AMIA, 2000.

[75] Z. Z. Hu, M. Narayanaswamy, K. E. Ravikumar, K. Vijay-Shanker, and C. H. Wu. Literature mining and database annotation of protein phosphorylation using a rule-based system. *Bioinformatics*, 21(11):2759–2765, June 2005.

[76] K. Humphreys, G. Demetriou, and R. Gaizauskas. Two applications of information extraction to biological science journal articles: Enzyme interactions and protein structures. In *Pacific Symposium on Biocomputing*, volume 2000, pages 502–513, 2000.

[77] C. Jacquemin. *Spotting and discovering terms through natural language processing*. The MIT Press, 2001.

[78] A. Jimeno-Yepes and R. Berlanga-Llavori. Study of named entity recognition in biomedicine: towards the refinement of ontologies. Technical report, Master Thesis, Universitat Jaume I, 2008.

[79] A. Jimeno-Yepes, R. Berlanga-Llavori, and D. Rebholz-Schuhmann. Ontology refinement for improved information retrieval. *Information Processing & Management: Special Issue on Semantic Annotations in Information Retrieval (to appear)*, 2009.

[80] A. Jimeno-Yepes, R. Berlanga-Llavori, and D. Rebholz-Schuhmann. Terminological cleansing for improved information retrieval based on ontological terms. In *Proceedings of the WSDM '09 Workshop on Exploiting Semantic Annotations in Information Retrieval*, pages 6–14. ACM, 2009.

[81] A. Jimeno-Yepes, R. Berlanga-Llavori, and D. Rebholz-Schuhmann. Topic template queries to enhance document retrieval. *ECIR workshop on Contextual Information Access, Seeking and Retrieval Evaluation*, 2009.

[82] A. Jimeno-Yepes, E. Jimenez-Ruiz, R. Berlanga-Llavori, and D. Rebholz-Schuhmann. Towards the enrichment of a biomedical ontology based on text mining. Technical report, European Bioinformatics Institute and Universitat Jaume I, 2007.

[83] A. Jimeno-Yepes, E. Jimenez-Ruiz, R. Berlanga-Llavori, and D. Rebholz-Schuhmann. Use of shared lexical resources for efficient ontological engineering. *Semantic Web Applications and Tools for Life Sciences*, 2008.

[84] A. Jimeno-Yepes, E. Jimenez-Ruiz, V. Lee, S. Gaudan, R. Berlanga-Llavori, and D. Rebholz-Schuhmann. Assessment of disease named entity recognition on a corpus of annotated sentences. *BMC Bioinformatics*, 9 Suppl 3, 2008.

[85] A. Jimeno-Yepes, P. Pezik, and D. Rebholz-Schuhmann. Information retrieval and information extraction in trec genomics 2007. In *Proceeding of Ninth Text REtrieval Conference (TREC-15). National Institute of Standards and Technology, Special Publication*, 2007.

[86] T. Joachims. A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization. In Douglas H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 143–151, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US.

[87] T. Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the Sixteenth International Conference on Machine Learning*, pages 200–209. Morgan Kaufmann Publishers Inc., 1999.

[88] R. Jones, R. Ghani, T. Mitchell, and E. Riloff. Active learning for information extraction with multiple view feature sets. *ECML-03 Workshop on Adaptive Text Extraction and Mining*, 2003.

[89] 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.

[90] L.R. Khan and F. Luo. Ontology construction for information selection. In *ICTAI '02: Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02)*, page 122, Washington, DC, USA, 2002. IEEE Computer Society.

[91] J. J. Kim and J. C. Park. Bioie: retargetable information extraction and ontological annotation of biological interactions from the literature. *Journal of Bioinformatics and Computational Biology*, 2(3):551–568, September 2004.

[92] J.J. Kim, P. Pezik, and D. Rebholz-Schuhmann. MedEvi: Retrieving textual evidence of relations between biomedical concepts from Medline. *Bioinformatics*, 24(11):1410, 2008.

[93] J. Köhler, K. Munn, A. Ruegg, A. Skusa, and B. Smith. Quality control for terms and definitions in ontologies and taxonomies. *BMC Bioinformatics*, 7:212, 2006.

[94] M.A. Krogel and T. Scheffer. Effectiveness of information extraction, multi-relational, and multi-view learning for prediction gene deletion experiments. In *BIOKDD*, pages 10–16, 2003.

[95] M.A. Krogel and T. Scheffer. Effectiveness of information extraction, multi-relational, and semi-supervised learning for predicting functional properties of genes. In *ICDM*, pages 569–572, 2003.

[96] V. Lavrenko and W.B. Croft. Relevance based language models. *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 2001.

[97] J.B. Lee, J.J. Kim, and J.C. Park. Automatic extension of gene ontology with flexible identification of candidate terms. *Bioinformatics*, 22(6):665–670, 2006.

[98] H. Li and N. Abe. Word clustering and disambiguation based on co-occurrence data. *CoRR*, cmp-lg/9807004, 1998.

[99] D. Lin. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics*, pages 768–774. Association for Computational Linguistics, 1998.

[100] Z. Liu and W.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.

[101] D. Losada and L. Azzopardi. An analysis on document length retrieval trends in language modeling smoothing. *Information Retrieval*, 11:109–138, 2008.

[102] A. Maedche and S. Staab. Discovering conceptual relations from text. In *Proceedings of the 14th European Conference on Artificial Intelligence (ECAI)*, pages 321–325, 2000.

[103] A. Maedche and S. Staab. Ontology learning for the semantic web. *IEEE Intelligent Systems*, 16(2):72–79, 2001.

[104] E.M. Marcotte, I. Xenarios, and D. Eisenberg. Mining literature for protein-protein interactions. *Bioinformatics*, 17(4):359–363, 2001.

[105] A. T. McCray, A. C. Browne, and O. Bodenreider. The lexical properties of the gene ontology (go). AMIA 2002, 2002.

[106] M. Missikoff, P. Velardi, and P. Fabriani. Text mining techniques to automatically enrich a domain ontology. In *Applied Intelligence 18*, pages 323–340. 2003 Kluwer Academic Publishers, 2003.

[107] M. Mitra, A. Singhal, and C. Buckley. Improving automatic query expansion. In *Research and Development in Information Retrieval*, pages 206–214, 1998.

[108] R. Navigli and P. Velardi. Automatic adaptation of wordnet to domains. In *Proceedings of 3rd International Conference on Language Resources and Evaluation*, 2002.

[109] R. Navigli and P. Velardi. An analysis of ontology-based query expansion strategies. In *Proc. of Workshop on Adaptive Text Extraction and Mining*, 2003.

[110] Roberto Navigli. Using cycles and quasi-cycles to disambiguate dictionary glosses. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 594–602, Athens, Greece, March 2009. Association for Computational Linguistics.

[111] J.Y. Nie and F. Jin. Integrating logical operators in query expansion in vector space model. In *Workshop on Mathematical/Formal Methods in Information Retrieval, 25thACM-SIGIR, Tampere, Finland*, volume 8, 2002.

[112] N.F. Noy and M.A. Musen. An algorithm for merging and aligning ontologies: automation and tool support. In *the Proceedings of the Workshop on Ontology Management at Sixteenth National Conference on Artificial Intelligence (AAAI-99), Orlando, FL.*

[113] D. Patterson, N. Rooney, V. Dobrynin, and M. Galushka. Sophia: A novel approach for textual case-based reasoning. In *IJCAI*, pages 15–20, 2005.

[114] J. Pearson. *Terms in Context.* Studies in Corpus Linguistics, 1. John Benjamins, Philadelphia, 1998.

[115] P. Pezik, A. Jimeno-Yepes, V. Lee, and D. Rebholz-Schuhmann. Static dictionary features for term polysemy identification. *Building and evaluating resources for biomedical text mining, LREC Workshop*, 2008.

[116] P. Pezik, A. Jimeno-Yepes, and D. Rebholz-Schuhmann. Lexical, terminological and ontological resources for biological text mining. In *Information Retrieval in Biomedicine: Natural Language Processing for Knowledge Integration. IGI Global Publishing*, 2009.

[117] V. Pillet, M. Zehnder, A.K. Seewald, A.L. Veuthey, and J. Petrak. Gpsdb: a new database for synonyms expansion of gene and protein names. *Bioinformatics*, 21(8):1743–1744, 2005.

[118] J.M. Ponte and W.B. 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.

[119] R. Porzel and R. Malaka. A Task-based Approach for Ontology Evaluation. In *ECAI Workshop on Ontology Learning and Population, Valencia, Spain*, 2004.

[120] W. Pratt, M.A. Hearst, and L.M. Fagan. A knowledge-based approach to organizing retrieved documents. In *AAAI/IAAI*, pages 80–85, 1999.

[121] W.M Pratt and H. Wasserman. QueryCat: Automatic Categorization of MEDLINE Queries. *Journal-American Medical Informatics Association*, 7:655–659, 2000.

[122] Y. Qiu and H.P. 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.

[123] D. Rebholz-Schuhmann, M. Arregui, S. Gaudan, H. Kirsch, and A. Jimeno-Yepes. Text processing through web services: Calling whatizit. *Bioinformatics*, 24, Number 2:296–298, 2007.

[124] D. Rebholz-Schuhmann, A. Jimeno-Yepes, M. Arregui, and H. Kirsch. Assessment of Modifying versus Non-modifying Protein Interactions. In *The Third International Symposium on Semantic Mining in Biomedicine*, 2008.

[125] D. Rebholz-Schuhmann, H. Kirsch, and G. Nenadic. IeXML: towards an annotation framework for biomedical semantic types enabling interoperability of text processing modules. *SIG BioLink, ISMB*, 2006.

[126] P. Resnik. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. In *International Joint Conference on Artificial Intelligence*, volume 14, pages 448–453. Lawrence Erlbaum Associates LTD, 1995.

[127] E. Riloff and R. Jones. Learning Dictionaries for Information Extraction by Multi-level Boot-strapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pages 1044–1049. The AAAI Press/MIT Press, 1999.

[128] E. Riloff and J. Shepherd. A corpus-based approach for building semantic lexicons. In Claire Cardie and Ralph Weischedel, editors, *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 117–124. Association for Computational Linguistics, Somerset, New Jersey, 1997.

[129] T.C. Rindflesch, J.V. Rajan, and L. Hunter. Extracting molecular binding relationships from biomedical text. In *Proceedings of the sixth conference on Applied natural language processing*, pages 188–195. Morgan Kaufmann Publishers Inc., 2000.

[130] B. Roark and E. Charniak. Noun-phrase co-occurence statistics for semi-automatic semantic lexicon construction. In *COLING-ACL*, pages 1110–1116, 1998.

[131] S.E. Robertson, S. Walker, and M.M. Hancock-Beaulieu. Large test collection experiments on an operational, interactive system: Okapi at TREC. *Information Processing and Management*, 31(3):345–360, 1995.

[132] 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.

[133] J. Rogers and A. Rector. The GALEN ontology. *Medical Informatics Europe (MIE 96)*, pages 174–178, 1996.

[134] B. Rosario and M.A. Hearst. Classifying semantic relations in bioscience texts. In *ACL*, pages 430–437, 2004.

[135] B. Rosenfeld, R. Feldman, M. Fresko, J. Schler, and Y. Aumann. Teg: a hybrid approach to information extraction. In *CIKM*, pages 589–596, 2004.

[136] C. Rosse, A. Kumar, J.L.V. Mejino Jr, D.L. Cook, L.T. Detwiler, and B. Smith. A Strategy for Improving and Integrating Biomedical Ontologies. In *AMIA Annual Symposium Proceedings*, volume 2005, page 639. American Medical Informatics Association, 2005.

[137] P. Ruch. Query translation by text categorization. In *Proceedings of the 20th international conference on Computational Linguistics*. Association for Computational Linguistics Morristown, NJ, USA, 2004.

[138] P. Ruch, R. Baud, and A. Geissbhler. Learning-free text categorization. *9th Conference on Artificial Intelligence, in Medicine in Europe*, 2003.

[139] B. Safar, H. Kefi, and C. Reynaud. OntoRefiner, a user query refinement interface usable for Semantic Web Portals . In *Applications of Semantic Web technologies to web communities, Workshop ECAI*, 2004.

[140] M. Sanderson. Word sense disambiguation and information retrieval. In *Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval*, pages 49–57, Dublin, IE, 1994.

[141] A.S. Schwartz and M.A. Hearst. A simple algorithm for identifying abbreviation definitions in biomedical text. *Pacific Symposium on Biocomputing*, 451:62, 2003.

[142] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[143] A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *Research and Development in Information Retrieval*, pages 21–29, 1996.

[144] M. Sintek, P. Buitelaar, and D. Olejnik. A formalization of ontology learning from text. In *Proceedings of the Workshop on Evaluation of Ontology-based Tools (EON2004) at the International Semantic Web Conference*, 2004.

[145] B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, LJ Goldberg, K. Eilbeck, A. Ireland, CJ Mungall, et al. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature Biotechnology*, 25(11):1251, 2007.

[146] B. Smith, W. Ceusters, B. Klagges, J. Khler, A. Kumar, J. Lomax, C. Mungall, F. Neuhaus, A. L. Rector, and C. Rosse. Relations in biomedical ontologies. *Genome Biology*, 6(5), 2005.

[147] B. Smith and C. Rosse. The role of foundational relations in the alignment of biomedical ontologies. *Medinfo*, 11(Pt 1):444–8, 2004.

[148] I. Spasić, S. Ananiadou, J. McNaught, and A. Kumar. Text mining and ontologies in biomedicine: Making sense of raw text. *Briefings in Bioinformatics*, 6(3):239–251, 2005.

[149] I. Spasić, D. Schober, S.A. Sansone, D. Rebholz-Schuhmann, D.B. Kell, and N.W. Paton. Facilitating the development of controlled vocabularies for metabolomics technologies with text mining. *BMC Bioinformatics*, 9(5):S5, 2008.

[150] R. Srikant and R. Agrawal. Mining generalized association rules. *Future Generation Computer Systems*, 13(2–3):161–180, 1997.

[151] G. Stoilos, G. Stamou, and S. Kollias. A string metric for ontology alignment. 4th International Semantic Web Conference (ISWC 2005), Galway, 2005.

[152] L. Venkata Subramaniam, S. Mukherjea, P. Kankar, B. Srivastava, V.S. Batra, P. V. Kamesam, and R. Kothari. Information extraction from biomedical literature: methodology, evaluation and an application. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 410–417, New York, NY, USA, 2003. ACM Press.

[153] C.A. Thompson, M.E. Califf, and R.J. Mooney. Active learning for natural language parsing and information extraction. In *Proceedings 16th International Conference on Machine Learning*, pages 406–414. Morgan Kaufmann, San Francisco, CA, 1999.

[154] J.D. Thompson, S.R. Holbrook, K. Katoh, P. Koehl, D. Moras, E. Westhof, and O. Poch. Mao: a multiple alignment ontology for nucleic acid and protein sequences. *Nucleic Acids Research*, 33:4164, 2005.

[155] J.I. Tsujii and S. Ananiadou. Thesaurus or logical ontology, which do we need for mining text? *Language Resources and Evaluation*, 39(1):77–90, September 2005.

[156] A. Vailaya, P. Bluvas, R. Kincaid, A. Kuchinsky, M. Creech, and A. Adler. An architecture for biological information extraction and representation. In *SAC '04: Proceedings of the 2004 ACM symposium on Applied computing*, pages 103–110, New York, NY, USA, 2004. ACM Press.

[157] C. J. van Rijsbergen. *Information retrieval*. Butterworths, London, 2 edition, 1979.

[158] E.M. 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.

[159] E.M. Voorhees. The philosophy of information retrieval evaluation. In *Revised Papers from the Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems*, pages 355–370. Springer-Verlag, 2002.

[160] E.M. Voorhees and C. Buckley. The effect of topic set size on retrieval experiment error. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 316–323. ACM New York, NY, USA, 2002.

[161] X. Wang and C.X. Zhai. A study of methods for negative relevance feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 219–226. ACM New York, NY, USA, 2008.

[162] D. Widdows. Unsupervised methods for developing taxonomies by combining syntactic and statistical information. In *HLT-NAACL*, 2003.

[163] W.J. Wilbur and W. Kim. Flexible Phrase Based Query Handling Algorithms. In *Proceedings of the ASIST Annual Meeting*, volume 38, pages 438–49, 2001.

[164] D.H. Wolpert and W.G. Macready. No free lunch theorems for optimization. *IEEE Trans. on Evolutionary Computation*, 1(1):67–82, 1997.

[165] L. Wong. PIES, a protein interaction extraction system. In *Proceedings of Pacific Symposium on Biocomputing*, volume 6, pages 520–531, 2001.

[166] J. Xu and W.B. 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.

[167] 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.

[168] A. C. Yu. Methods in biomedical ontology. *Journal of Biomedical Informatics*, 39(3):252–266, June 2006.

[169] H. Yu and E. Agichtein. Extracting synonymous gene and protein terms from biological literature. In *ISMB (Supplement of Bioinformatics)*, pages 340–349, 2003.

[170] S. Zhang and O. Bodenreider. Investigating implicit knowledge in ontologies with application to the anatomical domain. In *Pacific Symposium on Biocomputing*, pages 250–261, 2004.

[171] J. Zobel. How reliable are the results of large-scale information retrieval experiments? In *Research and Development in Information Retrieval*, pages 307–314, 1998.