Exploiting Semantic Annotations for Open Information Extraction: an experience in the biomedical domain

Victoria Nebot and Rafael Berlanga

Departamento de Lenguajes y Sistemas Informaticos,

Universitat Jaume I, 12071, Castellón, Spain

Abstract.

The increasing amount of unstructured text published on the Web is demanding new tools and methods to automatically process and extract relevant information. Traditional information extraction has focused on harvesting domain-specific, pre-specified relations, which usually requires manual labor and heavy machinery. Especially in the biomedical domain the main efforts have been directed towards the recognition of well-defined entities such as genes or proteins, which constitutes the basis for extracting the relations between the recognized entities. The intrinsic features and scale of the Web demand new approaches able to cope with the diversity of documents, where the number of relations is unbounded and not known in advance. This paper presents a scalable method for the extraction of domain-independent relations from text that exploits the knowledge in the semantic annotations. The method is not geared to any specific domain (e.g., protein-protein interactions, drug-drug interactions, etc.) and does not require any manual input or deep processing. Moreover, the method uses the extracted relations to compute groups of abstract semantic relations characterized by their signature types and synonymous relation strings. This constitutes a valuable source of knowledge when constructing formal knowledge bases, as we enable seamless integration of the extracted relations with the available knowledge resources through the process of semantic annotation. The proposed approach has successfully been applied to a large text collection in the biomedical domain and the results are very encouraging.

Keywords: Unsupervised IE; Relation extraction; Semantic annotation; Biomedical domain

Received Mar 26, 2012
Revised Sep 13, 2012
Accepted Oct 28, 2012

1. Introduction

Building and maintaining knowledge bases from both structured and unstructured text has been an active research field during the last years. Some efforts include *opencyc.org*[1], *dbpedia.org* [4], *yago* [47], *stat-snowball* [56] and *freebase.com*[2]. Several areas such as question answering, information extraction and textual entailment can benefit enormously from this factual knowledge.

The biomedical domain has attracted a lot of attention among the information extraction (IE) community due to the huge amount of scientific literature available (e.g., PubMed[3]). This scenario is very attractive, as most important knowledge still remains implicit in unstructured text, hindering its use and automatic exploitation by other applications. Moreover, the availability of already existing domain resources and databases as well as the standardization efforts in the biomedical community (e.g., *MeSH* [34], *SNOMED CT* [46], *UMLS* [10]) open interesting opportunities to integrate and augment the already existing resources with new knowledge automatically extracted from the literature.

Relation extraction (RE), which consists in finding associations between recognized entities within a text chunk, has recently found increasing interest among the IE community. The aim is to turn unstructured textual information into a machine-processable, structured form. This trend is in tune with the principles of the emergent Linked Data[4] philosophy and the Semantic Web [9].

However, the majority of RE approaches applied to the biomedical domain are aimed at extracting high precision relations about only a small, pre-defined set of specific relations of interest (e.g., protein-protein relationships, drug-disease relationships, etc.) [43, 1, 35]. Moreover, they require either hand-crafted extraction patterns or hand-labeled training data. These requirements make the existing RE methods difficult to scale and limit the extraction process. The recent open information extraction paradigm, Open IE [5, 6], has been successfully applied to extract general unrestricted knowledge in a Web environment. It attempts to overcome the previous limitations by providing a method which is relation independent and does not require labeled training data. However, the method has other important limitations that would make difficult to successfully apply it to the biomedical domain. Firstly, the extracted relations are not canonical, in the sense that they do not refer to well-defined entities. We consider a requirement to extract canonical relations that refer to entities in public knowledge resources (i.e., thesaurus, ontologies, etc.) as an enabling tool to realize the Semantic Web. Another requirement is imposed by the nature of the extraction process. Since the aim is to extract non-targeted unspecified relations, the granularity and semantics of the discovered relations needs to be taken care of so that synonymous relation strings are grouped under the same abstract relation. Synonym resolution for relations has been addressed by the NLP community under the broader field of *paraphrase discovery*, which includes not only synonymy between lexical items but also other forms such as hyperonymy. Many of the methods for determining the synonymy between two strings have made use of the well-known distributional similarity measures [28].

[1] `http://opencyc.org/`
[2] `http://freebase.com`
[3] `http://www.ncbi.nlm.nih.gov/pubmed/`
[4] `http://linkeddata.org`

Inspired by the same principles claimed in Open IE and the previous requirements we propose an unsupervised and scalable method for relation extraction and relation synonyms identification based on semantic annotation of textual sources. Given a text collection and a knowledge resource, a semantic annotation tool links the detected entities in the text to the concepts in the knowledge resource. This way, entities are leveraged with their semantics, which can be later used to enhance the relation extraction process. In a recall oriented phase, we apply a set of lexico-syntactic patterns over the annotated text to extract candidate relations. Then, an efficient clustering algorithm is proposed to group synonymous relation strings denoting abstract semantic relations. The synonymy between two relation strings is calculated with a probabilistic model based on the semantic types of the relation arguments provided by the semantic annotations of the entities. As a result, we obtain clusters of abstract semantic relations characterized by their signature types (i.e., domain and range semantic types) and containing the synonymous relation strings that best represent the abstract relation. Finally, the relation instances (i.e., concrete binary relations) are associated to the corresponding abstract semantic relation.

We summarize the contributions of the paper as follows:

- We use semantic annotation as the main foundation to identify and characterize binary relations in unstructured text.
- We propose a probabilistic model to measure relation synonymy which is semantics-aware. The model takes into account the distributions of the semantic types of the relation arguments.
- We propose an efficient clustering method to automatically discover groups of relation strings denoting abstract semantic relations with absolutely no human intervention at the Web scale.
- We apply the proposed method to a set of texts from the biomedical domain and demonstrate its scalability, assess the quality of the discovered clusters of semantic relations and suggest directions for future work.

The remainder of the paper is organized as follows. In Section 2, we discuss related research efforts. We present an overview of the method in Section 3. Sections 4, 5 and 6 elaborate on the different components of the method, the semantic annotation process, the pattern extraction and the discovery of abstract semantic relations, respectively. We evaluate the proposed approach in Section 7 and conclude with a discussion in Section 8.

2. Related Work

The approach proposed in this paper is related to work in several fields: information extraction, and in particular the subtask of relation extraction, and synonym resolution discovery. In this section we survey methods proposed in these disciplines both in the general and the biomedical domain.

2.1. Relation extraction

Common approaches for RE in the biomedical domain use pattern and rule-based [1], co-occurrence-based [24] and machine learning-based [43] methods. These

methods usually apply different natural language processing (NLP) techniques ranging from shallow parsing (e.g., POS tagging) to dependency parsing and deep linguistic parsing [50, 36, 14]. In our setting, we cannot afford to apply expensive and sophisticated parsing techniques as scalability is one of the main requirements.

Pattern and rule-based methods [1, 35, 23, 51, 22, 39] usually require human effort and intervention for updating and customizing patterns and rules to each application scenario. This issue is alleviated in [25], where the authors propose an inductive logic programming framework to learn the rules. Bootstrapping methods were proposed to learn both semantic lexicons and extraction patterns [42, 17]. The bootstrapping methods [48, 33] have been enhanced with reasoning capabilities to learn new constraints and rules and achieve both high precision and high recall.

Supervised machine learning techniques have also been successfully applied to train relation classifiers on human annotated texts in the biomedical domain [19, 12, 29]. They view the RE process as a classification problem, where the task consists in finding out if a particular relation holds between two entities in a sentence. Examples of supervised methods using kernels to encode lexical and syntactic features include [19, 29]. Other supervised approaches model the problem of RE as a sequence labeling problem and apply Markov logic and conditional random fields to identify the relations between two entities [43, 12]. The need of hand-labeled training examples of these approaches makes it difficult to scale to heterogeneous and large environments such as the Web. On the other hand, some of these methods address the identification of relations that are implicit in the text and cannot be captured by using lexico-syntactic patterns. This is a different problem from the one treated in this paper and is out of our scope as it requires different techniques.

The majority of the previous methods address the problem of RE in specific biomedical sub-domains that are of limited scope and require the set of target relations beforehand. However, we do not attempt to populate a given target relation, but rather discover any kind of relation relevant to the collection. Further, our approach is able to automatically obtain brand-new relations as they appear over time. The previous approaches also require some form of human intervention such as annotated training data or extraction patterns, as opposed to our method, which is fully unsupervised.

In this paper we focus on open IE, which attempts to extract domain-independent unknown relations. Although some research work has addressed the problem of open IE to extract general knowledge, few approaches have considered performing open IE in the biomedical domain[32]. Preemptive IE [45] is the precursor of open IE, as they describe an approach to "unrestricted relation discovery". Given a collection of news documents, they first cluster them based on pairwise vector-space clustering. Within each cluster they apply heavy linguistic machinery and additional clustering to group entities based on documents clusters. The computational cost of the approach makes it difficult to scale to a web environment. The most salient research in this context is *TextRunner* [53, 5, 6], which introduced the open IE paradigm. Although *TextRunner* is able to extract unknown relations at the Web scale, it does not harvest knowledge facts, but rather triples where both the relation and the arguments are simply non-canonical strings without any semantics attached to them. This makes the re-use of the extracted triples by other applications rather difficult and hinders the population of existing knowledge resources, which is one of the main objec-

tives to realize the Semantic Web. Other recent approaches that focus on open IE [56, 11] also suffer from the same issue. The work in [11] is related to ours since they propose a two-step procedure for relation extraction using clustering techniques. They first generate entity pairs and shallow lexical-syntactic patterns for the pairs from a given text corpus. Then a sequential co-clustering is performed to find clusters for entity pairs and lexical-syntactic patterns iteratively. The clustering of the entity pairs is not necessary in our approach thanks to the semantic annotation. Moreover, the dimensions of the input matrix and the complexity of the clustering algorithm make the approach difficult to scale to the Web.

2.2. Synonym resolution

The task of finding synonyms for the extracted relations is usually known as synonym resolution. Several methods for determining the synonymy between two relations have used the well-known *distributional similarity* metrics [28]. These metrics are based on the Distributional Hypothesis that says that "Similar objects appear in similar contexts" [20]. Therefore, they calculate the synonymy between two relations by comparing the arguments with which they occur. Several methods have been proposed [30, 31, 52, 49, 54], which differ in the representation of the predicates, the extracted features and the function used to compute the similarity of the feature vectors. To be effective, distributional metrics must rely on some weighting scheme over the relation features. The most adopted one is the pointwise mutual information (MI), which requires global counts such as $|(s, r, *)|$ and $|(*, r, o)|$, which are quite expensive to compute for large data sets.

Next, we explain the measure proposed by Lin and Pantel [31] as it is representative of the distributional similarity measures. They represent a predicate as a binary template $(relation, X, Y)$, where X and Y are the arguments of $relation$. For each binary template they compute two sets of features F_x and F_y, which are the words that instantiate the arguments X and Y, respectively, in a large corpus. Given a template t and its feature set for the X variable F_x^t, every $f_x \in F_x^t$ is weighted by the pointwise mutual information between the template and the feature: $w_x^t(f_x) = log\frac{Pr(f_x|t)}{Pr(f_x)}$, where the probabilities are computed using maximum likelihood over the corpus. Given two templates u and v, the similarity for the variable X is computed in the following way:

$$Lin_x(u, v) = \frac{\sum_{f \in F_x^u \cap F_x^v} w_x^u(f) + w_x^v(f)}{\sum_{f \in F_x^u} w_x^u(f) + \sum_{f \in F_x^v} w_x^v(f)}$$

The measure is computed analogously for the variable Y and the final distributional similarity score, in their $DIRT$ system, is the geometric average of the scores for the two variables:

$$DIRT(u, v) = \sqrt{Lin_x(u, v) Lin_y(u, v)}.$$

As previously mentioned, this measure requires global computations that are expensive for large data sets. This and other similarity measures will be compared to our synonymy model in the experimental section.

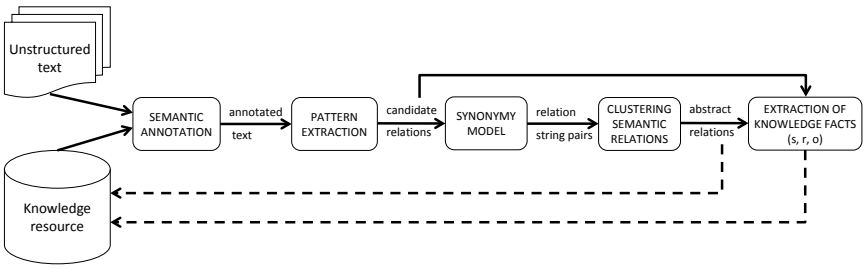

Fig. 1. Components of the proposed method.

3. Method Overview

The proposed method for scalable and domain-independent relation extraction is structured in a series of components which are independent of each other. This offers a great flexibility as different implementations of the components can be provided. The overall architecture of the method is illustrated in Figure 1. We consider a text collection as input to extract semantic relations, and the availability of a domain knowledge resource. Next, we explain each of the components of the method in detail:

1. *Semantic annotation*: We assume the availability of a semantic annotation tool able to map recognized entities in the text to concepts in the knowledge resource. In the following section we elaborate on the input and output requirements of the semantic annotation phase and explain the annotation tool used in our implementation.
2. *Pattern extraction*: This component is in charge of extracting candidate relations, that is, triples of the form $< s_1, r_s, s_2 >$ where r_s is a relation string representing the predicate of the relation and s_1 and s_2 are the arguments of the predicate. In our implementation the pattern extraction is based on the semantic annotations. Therefore, in the extracted candidate relations s_1 and s_2 are sequences of semantic annotations and r_s is a relation string represented by a simple lexico-syntactic pattern. This component is further detailed in Section 5. Notice that the order of the first two components can be altered, that is, the pattern extraction could be performed prior to the semantic annotation process as long as the provided extractor does not use the semantic annotations as reference to extract candidate relations.
3. *Synonymy model*: This component implements the synonymy model used to determine the similarity between two predicates. We implement a statistical model that takes the candidate relations $< s_1, r_s, s_2 >$ and calculates the synonymy probability of generating any two relation strings (r_1, r_2) based on the semantic types (T_d, T_r) associated to the entities acting as head of the sequences of semantic annotations s_1 and s_2. As a result, we obtain a list of relation string pairs (r_1, r_2) ordered by this probability. This phase is further detailed in Section 6.1.
4. *Clustering abstract semantic relations*: This component is in charge of group-

ing the relation candidates that are similar based on the synonymy model calculated previously. We implement an efficient clustering algorithm over the previous relation string pairs (r_1, r_2) to group synonymous relation strings. The clustering algorithm is semantics-aware as it takes into account the distribution probabilities of the semantic types of the arguments of the candidate relations. As a result, the surface relation strings are grouped into clusters of abstract semantic relations, which are characterized by the representative relation string and its signature type (the semantic types of the arguments). This phase is further detailed in Section 6.2.

5. *Extraction of knowledge facts*: This component associates the relation instances to the abstract semantic relation whose signature is compatible. Both the abstract semantic relations and the relation instances (i.e., facts) can be seamlessly integrated into the knowledge resource.

4. Semantic Annotation

In general terms, semantic annotation is conceived as the process of discovering and assigning to the recognized entities in the text links to their semantic descriptions, which are usually defined in a knowledge base [26, 40]. Since the main goal of extracting relations is to provide a comprehensive knowledge base of facts about named entities, their semantic classes and their mutual relations, we believe semantic annotation is crucial in the RE process to overcome heterogeneity and integration issues. Ideally, a semantic annotation must be explicit, formal, and unambiguous. These three properties enable machine understanding, and annotating with respect to an ontology makes this possible. A typical semantic annotation process includes three components. First, a knowledge resource or ontology must be available. Second, a data instance recognition process discovers instances of interest in target documents based on the ontology. Third, an annotation generation process creates the annotated documents.

With the proliferation of the Web of Data and initiatives such as the Linked Data project, which promotes a series of best practices to publish and link entities across the Web in a machine understandable way, many knowledge resources ranging from lexicons, terminologies and thesauri to expressive ontologies, are publicly accessible and ready to be used for annotation purposes. Some examples include *dbpedia*, *yago*, *freebase* and *schema.org*[5]. Specially in the biomedical domain we can find several lexical/ontological specialized resources such as *MeSH*, *SNOMED* and *UMLS* among others. For the instance recognition process, the available tools range from simple dictionary-based approaches, to more sophisticated NLP approaches that use NER tools, POS tagging, dependency parsing, etc. Some examples include *DBpedia Spotlight*[6], *The Wiki Machine*[7], *AlchemyAPI*[8] and *Open Calais*[9], for annotating general-purpose entities, and *MetaMap*[3] or *Whatizit*[38] for annotating biomedical entities.

The following definitions formalize the notion of semantic annotation. The examples show semantic annotations performed by our semantic annotation tool

[5] http://schema.org
[6] http://dbpedia.org/spotlight
[7] http://thewikimachine.fbk.eu
[8] http://www.alchemyapi.com/
[9] http://www.opencalais.com/

CMA [8], which uses *UMLS* as reference knowledge resource. Details about *UMLS* and this particular semantic annotation tool, as well as other alternatives are discussed in the remainder of the section.

Definition 4.1. Given a knowledge base *KB, and an input set* $S \subseteq \Sigma^*$ *of sequences over tokens from the alphabet* Σ*, a semantic annotation* e is a pair $< c, W >$ where $c \in concepts(KB)$ and $W \in \Sigma^*$ is a sequence $w_1 w_2 ... w_i$ such that there exists a mapping from c to a subset $W' \subseteq W$ (denoted by the function $m_{sa}(c)$) provided by the semantic annotation tool.

Next we show a semantic annotation that maps the concept *C0153876*, whose string name is "acute lymphocytic leukemia", to the tokens *acute lymphocytic leukemia*. This particular semantic annotation tool takes advantage of the classification of concepts into semantic types and semantic groups and shows this information, that is, the knowledge resource (*UMLS*), the semantic type of the concept (*T191*, which corresponds to *Disease and Syndrome*) and the semantic group (*DISO*, which corresponds to *Disorder*). Notice that the mapped tokens are referenced by their position number in the sequence, allowing mappings to non-consecutive tokens.

```
<e id="UMLS:C0153876:T191:DISO::1,2,3>
<w id="1">acute</w> <w id="2">lymphocytic</w>
<w id="3">leukemia</w> </e>.
```

This other example shows a semantic annotation where the mapped tokens are not consecutive.

```
 <e id="UMLS:C0003313:T116;T129:PRGE::1,3,4">
 <w id="1">circulating</w> <w id="2">soluble</w>
 <w id="3">immune</w>
 <w id="4">complexes</w> </e>
```

Definition 4.2. Given a knowledge base KB, an input set $S \subseteq \Sigma^*$ of sequences over tokens from the alphabet Σ and the set of semantic annotations E calculated by the semantic annotation tool, an ambiguous semantic annotation e is a pair $< c, W >$ such that there exists another semantic annotation $e' =< c', W' > \in E$ where $m_{sa}(c) = m_{sa}(c')$.

A semantic annotation is ambiguous if more than one concept has been assigned the exact same subset of tokens. In the following example, the string *Plasmodium falciparum* has been annotated with three different concepts that belong to different semantic groups (i.e., chemicals, living beings and proteins-genes).

```
<e id="UMLS:C0487378:T129:CHED::1,2|
       UMLS:C0032150:T204:LIVB::1,2|
       UMLS:C0369855:T116;T129:PRGE::1,2">
<w id="1">Plasmodium</w> <w id="2">falciparum</w></e>
```

4.1. UMLS as biomedical knowledge resource

The Unified Medical Language System (*UMLS*)[10] is probably the most comprehensive knowledge resource in the biomedical domain. The version 2010AA has more than 2.2 million concepts over 150 source vocabularies. *UMLS* includes

[10] UMLS: `http://www.nlm.nih.gov/research/umls/`

three resources: the Metathesaurus, the Semantic Network[11], and the Specialist Lexicon. The first one consists of several thesauri (e.g., *FMA*, *MeSH*, etc.) which contain Atoms (the distinct concepts within each source). Each Atom is assigned to one and only one Concept, whose identifier is called CUI. The semantic annotation tools usually work at the level of CUIs. The Semantic Network provides a consistent categorization of all concepts represented in the Metathesaurus and provides a set of useful relationships between these concepts. It contains 133 semantic types linked by "is a" relationships. There are also 54 kind of non-hierarchical relationships among these semantic types, e.g., $causes(Virus, Disease)$. A concept of the Metathesaurus can be assigned to one or more semantic types in the Semantic Network. The *UMLS* semantic groups[12] is an even smaller and coarser-grained set of semantic type groupings. It contains 15 groups.

For all the previous features, UMLS is a good knowledge resource to use for semantic annotation.

4.2. Biomedical semantic annotation tools

Most previous work in semantic annotation in the biomedical domain has been restricted to the identification of protein and gene names [21, 55]. Recently, the focus has shifted from individual genes and proteins to the identification of entire biological systems, disease names, etc.

MetaMap [2, 3] was one of the first tools for mapping biomedical terms in free text to UMLS concepts. It allows partial matching between text spans and lexical forms by means of significant linguistic analysis followed by mapping construction from intermediate results. One of the limitations of MetaMap is that it is very tightly coupled with the UMLS, making the use of custom dictionaries outside of UMLS non-trivial. Moreover, precision is usually low compared to dictionary look-up approaches and it suffers from scalability issues.

Recently, there have been a number of tools such as Whatizit [38], Mgrep [16] and CONANN [41] that also perform semantic annotation of concepts. Dictionary look-up approaches such as Whatizit and Mgrep allow fast execution and scalability by finding in the documents each text span that exactly matches some lexical forms of the terminological resource. Although these approaches exhibit good precision numbers, their recall is usually low. CONANN is an online biomedical concept annotator whose philosophy of using candidate concepts and a score based on inverse document frequency is close to the CMA tool used in this paper.

The interest in semantic annotation of biomedical entities is such that initiatives such as CALBC (Collaborative Annotation of a Large Biomedical Corpus) [15] have been set up with the goal of providing a silver standard corpus (SSC) of annotated biomedical entities to the community. These annotations are the result of an agreement between the participants annotations.

It must be noticed that these tools have been designed for identifying entity mentions but not for extracting relations between them. As a consequence, these tools do not care for the precise boundaries of the entities participating in each identified relation, which are necessary to determine the semantic type of

[11] UMLS SN: `http://semanticnetwork.nlm.nih.gov/`
[12] UMLS Semantic Groups: `http://semanticnetwork.nlm.nih.gov/SemGroups/`

Metamap annotations
$< the\ isolated\ lung\ strip\ of\ the\ cat >^{CONC,ANAT,LIVB}$ is $< described\ for\ investigating >^{CONC} < the\ direct\ effect\ of\ drugs >^{CONC} < on\ the\ smooth\ muscle >^{ANAT} < of\ the\ peripheral\ airways >^{CONC,DEVI} < of\ the\ lung >^{ANAT}$
CALBC SSCII annotations
the isolated lung strip of the $< cat >^{LIVB}$ is described for investigating the direct effect of drugs on the smooth muscle of the peripheral airways of the $< lung >^{ANAT}$.
CMA annotations
the isolated $< lung >^{ANAT}$ strip of the $< cat >^{LIVB}$ is described for investigating the direct effect of drugs on the $< smooth\ muscle >^{ANAT}$ of the peripheral $< airways >^{ANAT}$ of the $< lung >^{ANAT}$.

Table 1. Annotation examples of MetaMap, and the CALBC annotations and CMA. The upper script denotes the semantic type assigned by the semantic annotation tool.

the relation arguments. For example, MetaMap maximizes the text spans that correspond to UMLS concepts, producing chunks that do not allow a proper identification of the relation arguments. On the other hand, more specific tools like protein/gene annotators usually produce very small text chunks, which may not reflect the true semantic types of the extracted relations. Table 1 shows a sentence annotated by MetaMap, in which practically all the sentence is annotated, the same sentence in the CALBC SSCII annotations, where only two isolated words have been annotated and the annotations performed by CMA.

4.3. CMA tool

Here we describe our semantic annotation tool Concept Mapping Annotation (CMA) [8]. Compared to other annotation alternatives, this tool offers a clean and simple approach and shows a good trade-off between performance and scalability. The tool has been successfully used in other scenarios, such us in [37], where they annotate the textual descriptions provided by catalogues of Life Science Web Services and align them with the user requirements. It is worth mentioning that this tool does not disambiguate annotations and can make errors. Indeed, the proposed method is aimed at capturing abstract relations despite some limitations in the annotation process.

The tool is based on information retrieval (IR) models. Thus, it measures the similarity between a given query (i.e., a text fragment) and each document of the collection (i.e., concept descriptions) in order to give a conceptual cover for the query. We have adopted the following information-theoretic similarity function:

$$sim(C,T) = max_{S \in lex(C)}(ratio(S,T))$$

$$ratio(S,T) = \frac{info(cw(S,T)) - missing(S,T)}{info(S)}$$

$$missing(S,T) = (info(S) - info(cw(S,T)))$$

where $info(S)$ measures the relevance of the words in the string S, and

$cw(S,T)$ is the set of words in common between the concept description S and the text fragment T. It is defined as follows:

$$info(S) = -\sum_{w \in S} log(P(w|KR))$$

The relevance of a set of words is measured by means of the estimated probability of each word within the whole knowledge resource (KR) lexicon (i.e., $P(w|KR)$). In this way, highly frequent words in the KR contribute little to the final score of the strings containing them. The final score $sim(C,T)$ is normalized (i.e., it ranges between 0 and 1). Notice that not all the words of the string S must appear in the text T, but just those that better discriminate the intended concepts. It is worth mentioning that this score does not require any parameter except an estimation of $P(w|KR)$ for each word w in the KR lexicon.

In our experiments, text fragments T are selected according to the relations to be extracted. More specifically, we have tested two different configurations: an ad-hoc chunker for identifying noun phrases around relations by using a POS-tagger, and the triples provided by the ReVerb[13] tool.

5. Pattern extraction

Biomedical literature is characterized by making use of sentences with complex structure and specialized vocabulary. This represents a challenge for the IE community. The use of long sentences with relative and conjunctive clauses complicates the matter and requires deep linguistic analysis of each sentence. The discovery of implicit relations in complex and long sentences is out of the scope of this paper, as it is a time consuming task usually addressed by machine learning approaches. In this paper, we focus on the discovery of explicit binary relations of the form $< subject, predicate, object >$. For that, we propose a set of lexico-syntactic patterns. We are aware of the variety of syntactical patterns used to express relations in the biomedical domain. These range from simple $< subject, predicate, object >$ patterns where the predicate is a verbal form (e.g., $< influenza, induces, asthma >$) to verb nominalizations and complex alternations both of verbs and nouns [13]. Table 2 shows some examples of the latter. The proposed patterns are not exhaustive and thus, do not capture all the previous linguistic and syntactical variations. We believe such patterns require different linguistic and deeper analysis from the one proposed in this paper. In any case, frequent verb nominalizations such as *binding, association, interaction, inhibition, phospholiration, etc.* happen to be annotated by our tool under the UMLS semantic groups PHEN (Phenomena) and PHYS (Physiology). Therefore, their identification for further extraction and analysis would be possible. We leave the treatment of such kind of patterns for future work.

The following section introduces the set of lexico-syntactic patterns proposed for the extraction of candidate relations. However, the loose coupling of each of the components of the method allows to use any other pattern extractor that adheres to the $< subject, predicate, object >$ format.

[13] ReVerb: `http://reverb.cs.washington.edu/`

Verb nominalizations semantically annotated
$< phosphorylation^{PHYS} >$ of $< Rb^{PRGE} >$ by the $< cyclin^{PRGE} >< D1/cdk4^{PRGE} >$
$< Cdc2^{PRGE} >< phosphorylation^{PHYS} >$ of $< nucleolin^{PRGE} >$
$< 125I - IGF - I^{CHED} >< binding^{PHYS} >$
$< DNA - binding^{CHED} >< assay^{PROC} >$
$< PP2A^{PRGE} >< association^{PHEN} >$ with $< Shc^{PRGE} >$
$< Herpesvirus^{LIVB} >< type2^{MISC} >< association^{PHEN} >$ with $< carcinoma^{DISO} >$
$< T - cell^{PRGE} >< interactions^{PHYS} >$
$< Gm^{PRGE} >< interaction^{PHYS} >$ in $< SLE^{DISO} >$
$< PGN - induced^{PRGE} >< inhibition^{PHYS} >$ of $< vaccinia\ virus\ replication^{PHYS} >$

Table 2. Examples of verb nominalizations. The upper script denotes the semantic type assigned by the semantic annotation tool.

5.1. Simple lexico-syntactic patterns

Although the biomedical literature uses especially complex structures, recent studies have shown that a high percentage of binary relationships are expressed in English sentences by a set of few lexico-syntactic patterns that can be detected with a shallow analysis (POS-tagging) [6]. We take advantage of this fact in order to gather from the annotated corpus a subset of candidate binary relationship instances (i.e., candidate relations) by using the set of lexico-syntactic patterns (LSP) shown in Table 5.1. Our key insight is that semantic annotations in a sentence almost always correspond to the arguments of the relations. Therefore, the lexico-syntactic patterns are looked for in between two semantic annotations. However, depending on the leniency of the semantic annotation tool used, the semantic annotations may be too restrictive to capture the complete meaning of the entity. To cope with this issue, we extend the definition of semantic annotation in order to completely capture the meaning of the arguments of the relations.

Definition 5.1. Given the set of semantic annotations E calculated by the semantic annotation tool, and the set P of English prepositions, we define a sequence of semantic annotations as a sequence $e_1 e_2 ... e_n$ where $e_i \in E \cup P$.

A sequence of semantic annotations is a subset of annotations appearing in consecutive order or separated by prepositions. In the following sentence, "*The effect of [cyclosporin A] on [dental caries] in [rats] monoassociated with [Actinomyces viscosus]*", where annotated entities appear between brackets, we identify s_1 =*cyclosporin A on dental caries in rats* and s_2 =*Actinomyces viscosus* as sequences of semantic annotations.

Finally, we define the candidate relations, which are the relation instances from which the groups of semantic relations and their type constraints are learned.

Definition 5.2. A candidate relation $f \in \Sigma^*$ is a triple $< s_1, r_s, s_2 >$ such that s_1 and s_2 are sequences of semantic annotations and r_s (the relation string) is an instance of a lexico-syntactic pattern.

Pattern	Examples
[E] verb [E]	[levamisole] activates [macrophages]
	[secretory phospholipase A2] induces [dendritic cell maturation]
[E] verb phrase [E]	[PAF] consistently inhibited [Killer cell]
[E] verb phrase + prep [E]	[polysacharide] was treated with [periodate]
	[Normal spleen cells] treated with [poly I:C]
	[neutrophilia] were induced by [LPS]
	[ATF-2] was inhibited by [HDAC3]
	[IFNs] are principally mediated by [GAF]
[E] prep + noun + prep [E]	[cytostatic drugs] in combination with [OK-432]
[E] to + infinitive [E]	[fibroblasts] to produce [growth factor(s)]
[E] neg-verb-phrase [E]	[haptens] does not inactivate [B lymphocytes]
[E] to be [E]	[Strongyloidiasis] is an [intestinal disease]

Table 3. Lexico-syntactic patterns (LSP) proposed

In the previous example, the candidate relation $f =< s_1, r_s, s_2 >$ is composed by s_1 =*cyclosporin A on dental caries in rats*, the relation string r_s=*monoassociated with* and s_2 =*Actinomyces viscosus*. We emphasize the importance of detecting sequences of semantic annotations as possible arguments for the relations. Without this, the candidate relation of the previous example would read s_1 =*rats*, r_s=*monoassociated with*, s_2 =*Actinomyces viscosus*, which does not reflect the exact meaning of the sentence. The semantic annotations acting as heads of s_1 and s_2 play a crucial role in identifying the semantic types of the discovered abstract semantic relations. Therefore, we use the function $head(seq) : S \rightarrow E$ to return the annotation acting as head of a sequence of annotations. Following the example, $head(s_1)$ =*cyclosporin A* and $head(s_2)$ =*Actinomyces viscosus*.

A candidate relation $f =< s_1, r_s, s_2 >$ is well-typed if $head(s_1)$ and $head(s_2)$ are unambiguous annotations. We will only deal with well-typed relations as they have a unique semantic type associated to its subject and object.

6. Discovering Abstract Semantic Relations

The main goal of this paper is to find semantic groups of relation strings (i.e., abstract semantic relations) which represent relations between entity types. In other words, given the set of extracted candidate relations $< s_1, r_s, s_2 >$ from the previous phase, we want to find groups of relation strings r_s that may be used in the same contexts with a similar purpose. Moreover, we want to characterize the groups with a signature type. The following definition formalizes the notion of abstract semantic relation.

Definition 6.1. An abstract semantic relation is a tuple $(< T_d, R, T_r >, Syn)$ such that $< T_d, R, T_r >$ is the signature, where T_d and T_r are semantic types for the domain and range of the relation, R is the representative name of the relation, and $Syn = \{r_s\}$ is the set of synonymous relation strings.

The different abstract semantic relations are discovered by clustering the candidate relations according to their joint synonymy probability, which depends on the usage of the relation strings in different contexts (i.e., with different entity

types). The following sections explain the statistical model used to measure the synonymy and the clustering algorithm.

6.1. Statistical model for synonymy

We say that two relation strings r_1 and r_2 are synonyms, denoted $r_1 \sim r_2$ if we can replace one string by the other without changing the meaning of the relation. As a same relation string can be applied over different domain and range types, synonymy should be defined in terms of the semantic types. That is, we must specify the semantic types (T_d, T_r) for the domain and range respectively, under which the synonymy property holds. For example, $to\_treat\_with \sim to\_inject\_with$ holds for the semantic types $(Human, Drug)$, but not in other contexts. We use the function $types((r_1, r_2))$ to refer to the semantic types.

Our approach to find synonymous relations consists in applying a statistical translation method, inspired by the information retrieval translation models of [27]. Basically, we want to estimate the joint probability of generating two relation strings for any pair of semantic types (T_d, T_r). The joint probability can be estimated as follows:

$$p(r_1 \sim r_2|(T_d, T_r)) \propto \sum_{(e_1,e_2)\in(T_d,T_r)} p(r_1|(e_1, e_2)) \cdot p(r_2|(e_1, e_2)) \cdot p((e_1, e_2))$$

This is achieved by first estimating the language model of each relation string r_s, $p(r_s|(e_1, e_2))$, where (e_1, e_2) is any pair of annotated entities in the corpus that has semantic types (T_d, T_r). These language models are just estimated through the relative frequency of observing each relation string with respect to each pair of annotated entities. From the candidate relations $< s_1, r_s, s_2 >$ we calculate the probabilities $p(r_s|(e_1, e_2))$ and $p((e_1, e_2))$ such that $e_1 = head(s_1)$ and $e_2 = head(s_2)$.

Notice that each pair (e_1, e_2) is an evidence of synonymy between the compared relation strings. Therefore, the number of pairs that should support the probability estimation must be statistically significant. For this purpose, in the experiments we establish a threshold for the synonymy probability based on the minimum number of entity pairs that is statistically significant.

Independently from the previous joint probability, if two relation strings, r_1 and r_2, have the same signature (T_d, T_r) and their heads share the same stem, they are considered synonymous and thus, a high probability is assigned. For example, the relation strings *localized in* and *colocalized in* are considered synonyms under the signature $(Protein, Protein)$ as they share the stem $localiz*$.

Due to errors in the annotations as well as in the pattern extraction process, it is possible to find some false associations between relation strings. One way to check if two relation strings are being used properly under the given context is to compare their distributions across all the pairs of semantic types (T_d, T_r), denoted as $p(r_s|(T_d, T_r))$. These distributions can be compared with the Kullback-Leibler divergence, as follows:

$$D_{KL}(r_1, r_2) = \sum_{\forall(T_d,T_r)} p(r_1|(T_d, T_r)) \cdot log\frac{p(r_1|(T_d, T_r))}{p(r_2|(T_d, T_r))}$$

Similarly to the previous models, probabilities $p(r_1|(T_d, T_r))$ are estimated by

calculating the relative frequency of the relation string across the pairs (T_d, T_r). In this case, these probabilities must be smoothed to avoid zero probabilities. Notice that pairs of relation strings that present a great divergence are likely to be wrong, and therefore, they are unlikely to be synonymous. As such, the divergence threshold is used to filter noise produced by both wrong annotations and extraction errors. As shown in the experiments, the more errors an annotated collection contains the more sensitive is the clustering algorithm to the divergence threshold. Thus, users can set this parameter depending on the quality of both the extracts and the intended results.

6.2. Clusters denoting abstract semantic relations

The abstract semantic relations as defined previously are characterized by a signature type $< T_d, R, T_r >$ and a set of tightly related relation strings Syn. In order to create the different clusters of abstract semantic relations we cluster the list of ordered relation string pairs (r_1, r_2) in descending order of synonymy probability as previously calculated, that is, $p(r_1 \sim r_2|(T_d, T_r))$.

Definition 6.2. A pair of relation strings (r_1, r_2) belongs to the abstract semantic relation $C = (< T_d, R, T_r >, Syn)$, i.e., $r_1 \in Syn$ and $r_2 \in Syn$, iff $types((r_1, r_2)) = (T_d, T_r)$ and $rep(C) \cap (r_1, r_2) \neq \emptyset$.

In the previous definition, $rep(C)$ is the representative relation string pair of the cluster, which happens to be the pair with greater synonymy probability. In order to generate the clusters of abstract semantic relations, we propose the greedy Algorithm 1. The algorithm requires a list of ordered relation string pairs (r_1, r_2). First, the algorithm filters out the pairs of relation strings with divergence greater than a threshold θ because these pairs are likely to be wrong synonymous strings. Then, the filtered list of relation string pairs is partitioned according to the semantic types such that each partition in $Part_{types}$ holds relation string pairs with the same signature. Each of these partitions will generate a set of clusters stored in C_{types}. Since the relation string pairs keep the order in each partition, the stronger synonymous pairs are processed first. For each pair, the algorithm either generates a new cluster and sets that pair as representative or the pair is appended to an existing cluster if it has a common relation string with the representative of such cluster. Notice that the intersection operator is not strict, as we use a lexical similarity measure to compare two relation strings. Each partition generates a set of clusters C_{types} sharing the same semantic types (T_d, T_r) that is appended to $Clusters$.

6.3. Analysis cost

In order to calculate the probabilities $p(r_1 \sim r_2|(T_d, T_r))$, each candidate relation $< s_1, r_s, s_2 >$ is processed and for each entity pair (e_1, e_2) such that $e_1 = head(s_1)$ and $e_2 = head(s_2)$ we keep the different relation strings in a list. Then, the relation strings of each list are taken by pairs (r_1, r_2) and the probabilities $p(r_1|(e_1, e_2))$, $p(r_2|(e_1, e_2))$ and $p((e_1, e_2))$ are computed using maximum likelihood over the corpus as follows: let N be the number of entity pairs (e_1, e_2) that occur with more than one relation string r_s. The probability $p((e_1, e_2))$ can be estimated as $\frac{1}{N}$. Let n_1 be the cardinality of a specific relation string r_s co-

occuring with (e_1, e_2) and n_2 be the total number of relation strings co-occuring with (e_1, e_2). The probability $p(r_s|(e_1, e_2))$ can be estimated as $\frac{n_1}{n_2}$. Therefore, the cost of estimating the probabilities is $O(N)$.

The cost of ordering the relation string pairs (r_1, r_2) is $O(MlogM)$ being M the number of relation string pairs.

Finally, the clustering algorithm is in the worst case linear with respect to the number of relation string pairs.

In contrast to distributional metrics, our approach does not require to build feature vectors nor global computations like mutual information. Indeed the computation of the proposed synonymy metric can be massively parallelized for dealing with very large data sets.

Algorithm 1 Generation of abstract relations

Require: LR: list of relation string pairs ordered by synonymy probability, θ: threshold for the relation divergence
Ensure: $Clusters$: a clustering of relation strings
  $Clusters = \emptyset$
  $LR' = \{(r_{i1}, r_{i2}) \in LR \mid D_{KL}((r_{i1}, r_{i2})) < \theta,\ 0 < i \leq |LR|\}$
  $Part_{types} = \{P_1, ..., P_m\}$ such that $\forall (r_{i1}, r_{i2}), (r_{j1}, r_{j2}) \in P_k\ types((r_{i1}, r_{i2})) = types((r_{j1}, r_{j2})),\ 0 < i, j \leq |P_k|,\ 0 < k \leq m$
  for all $P_i \in Part_{types}$ **do**
    $C_{types} = \emptyset$
    for all $(r_{i1}, r_{i2}) \in P_i$ **do**
      if $\exists C_i \in C_{types}$ such that $(r_{i1}, r_{i2}) \cap rep(C_i) \neq \emptyset$ **then**
        append (r_{i1}, r_{i2}) to C_i
      else
        create new cluster $C = \{(r_{i1}, r_{i2})\}$
        $rep(C) = (r_{i1}, r_{i2})$
        $C_{types} = C_{types} \cup C$
      end if
    end for
    $Clusters = Clusters \cup C_{types}$
  end for
  return $Clusters$

7. Experiments

We have performed several experiments to test our general method for relation extraction and synonymy resolution based on semantic annotation. The first experiment evaluates the quality of the clusters of synonymous relations obtained by our method by trying with different configurations of semantic annotation and pattern extraction. In this experiment we also analyze the sensitivity of each configuration to the divergence threshold. The second experiment compares the quality of the clusters using different known distributional similarity measures. We also found interesting to compare the extracted semantic relations to those in the UMLS SN, as it is a reference knowledge resource in the biomedical domain. Finally, we show some examples of the extracted clusters of semantic relations and the facts (i.e., relation instances) classified.

7.1. Setup

The selected application scenario for evaluating the method is the biomedical domain, as there are huge amounts of unstructured text containing implicit knowledge available and also several knowledge resources describing biomedical entities. In fact, this research was partly motivated by the CALBC initiative, which addresses the automatic generation of a very large, shared text corpus annotated with biomedical entities. For that, they set up two challenges. In the second, they provided the participants with two corpora of Medline immunology-related abstracts to be annotated. The small corpus contains about 175 thousand abstracts and the big one contains about 714 thousand. The goal is to provide a silver standard corpus useful for text-mining tasks. We believe that such amount of semantically annotated documents constitutes an invaluable source of knowledge and a great opportunity to advance the automatic extraction of biomedical relations. Therefore, the experiments about open relation extraction were carried out on the set of 714 thousand documents, which we refer with the name of *CALBC corpus*. As the component-based structure of our method allows flexibility, we provide different configurations for each of the components (i.e., semantic annotation, pattern extraction and synonymy model) for the input set of documents of the CALBC corpus. Next, we explain each of the configurations:

- **CALBC SSCII + LSP (C1)**: This configuration is the result of using the annotated silver standard corpus of the second challenge of the CALBC initiative for the semantic annotation component and our LSP patterns for the pattern extraction phase. The semantic annotation process is the result of the harmonization of the semantic annotations of the participants on a 4-votes based agreement.
- **CMA + LSP (C2)**: This configuration is the result of using the semantic annotation tool CMA and the LSP in the pattern extraction phase to the CALBC corpus.
- **ReVerb + CMA (C3)**: This configuration is the result of applying the CMA annotation tool to the triples extracted by ReVerb, which consists of text chunks with the implicit subject-predicate-object structure. ReVerb is a triple extractor that identifies relation strings that satisfy some syntactic and lexical constraints mainly based on POS tags, and then finds a pair of noun phrase arguments for each identified relation string. Further details can be found in [18]. In this configuration, we switch the order of the semantic annotation and the pattern extraction phases because the smart chunking performed by ReVerb can ease the annotation process.

Table 4 summarizes the main features of the three configurations used. The first two columns show the semantic annotation tool and the patterns used. For C1, the semantic annotation tool refers to the already annotated CALBC corpus that is the result of the harmonization process between the annotations of different systems. The third column shows the number of obtained semantic annotations. The fourth and fifth columns show the number of candidate relations of the form $< s, r, o >$ and the number of filtered relations. The set of filtered relations is obtained by discarding candidate relations that are ambiguous (i.e., relations whose subject or object annotations are ambiguous) and also discarding vague relations (i.e., candidate relations whose subject, object or predicate

	Sem. Ann.	Patterns	# Ann.	# Cand. rels.	# Filtered	Cov. (%)
C1	CALBC_{SSCII}	LSP	10,304,172	173,044	29,803	17.22
C2	CMA	LSP	21,929,106	977,717	262,785	26.87
C3	CMA	ReVerb	21,772,578	4,607,027	633,446	13.74

Table 4. Several configurations for the semantic annotation and pattern extraction components over the the CALBC corpus input dataset.

strings have low inverse document frequency[14]). Finally, the last column indicates the percentage of filtered relations w.r.t. the initially extracted candidate relations.

MetaMap was one of the participants in the CALBC competition, but only performed the annotation over the small collection, which contains around 175 thousand abstracts. Due to its high recall, MetaMap annotated almost all the text, and consequently it was not possible to extract a significant number of triples (around 4000 triples) to evaluate it.

We have manually set up a gold standard (GS) to account for synonym groups of relation signatures that can be derived from the UMLS Semantic Network and are present in the evaluated collection. We mainly follow the evaluation methodology proposed in [54], but also regarding the semantic types of the relations. More specifically, for each pair of semantic types expressed in the UMLS Semantic Network, we have selected relation strings that are included in other gold standards (e.g., protein-protein, drug-drug, protein-disease interactions) along with those that appear frequently in our dataset under the corresponding signature. Afterwards, we have manually clustered synonymous strings into 249 groups [15]. As in [54], we use precision (P), recall (R) and F-score (F1) to measure the overlap between the best matches between the system-generated clusters and the GS groups. These matches only consider non-singleton clusters that have at least one relation string appearing in the GS. Finally, macro-averaging is used for calculating the global scores of these measures. In the evaluation we also use the coverage (Cov) of the candidate relations, which is the percentage of candidate relations covered by the system-generated clusters (non-singletons). Usually, the greater the coverage the more difficult it is to find synonym clusters. The GS has a coverage of 32% with 249 non-singleton clusters.

7.2. Evaluation of the clusters of semantic relations

The clusters of abstract semantic relations group relation strings that are synonymous under the same context (i.e., argument types). Our first experiment evaluates the obtained clusters for the different configurations proposed for the CALBC corpus. We set the synonymy probability threshold of a pair of relation strings to 1^{-5}. The divergence threshold θ between pairs of relation strings is used to filter noise produced by both wrong annotations and extract errors. Table 5 shows the evaluation for the three configurations with different settings of the divergence threshold.

[14] We have estimated it with the Wikipedia 2007 snapshot.
[15] `http://krono.act.uji.es/Links/datasets/GSsynonyms.txt`

Configuration	Metrics	Div. threshold θ				
		0.0001	0.001	0.01	0.1	1
C1	P	0.99	0.98	0.80	0.56	0.53
	R	0.72	0.72	0.67	0.68	0.70
	F1	0.79	0.79	0.65	0.49	0.48
	Clusters	264	267	429	778	829
	Avg. cluster size	3.73	3.75	4.05	5.37	5.88
	Coverage (%)	27.02	27.52	38.95	62.13	65.42
C2	P	0.98	0.98	0.96	0.93	0.93
	R	0.57	0.57	0.58	0.58	0.58
	F1	0.64	0.64	0.64	0.63	0.63
	Clusters	3250	3251	3314	3350	3352
	Avg. cluster size	7.35	7.28	7.15	8.04	8.43
	Coverage (%)	34.69	34.70	37.37	39.10	39.29
C3	P	1.00	1.00	0.97	0.95	0.95
	R	0.55	0.55	0.56	0.57	0.57
	F1	0.64	0.65	0.64	0.63	0.63
	Clusters	4432	4440	4522	4564	4564
	Avg. cluster size	4.63	4.64	4.68	4.72	4.72
	Coverage (%)	32.19	32.52	34.70	36.22	36.25

Table 5. Evaluation of the clusterings obtained with different configurations of the initial data set and different settings of the divergence threshold θ.

The effect of the divergence threshold over the three configurations is interesting. The general tendency is that as the divergence increases, the clustering algorithm is less restrictive, allowing groups of relation strings that might not be synonymous because of its high divergence. The algorithm tends to extract more relations covered by more clusters, as both the number of clusters and the average cluster size increase. This favors both recall and coverage but affects the precision of the synonymous relation strings captured by the clusters. Although the three configurations follow the previous tendency, we observe that C1 is noticeably more sensitive to the divergence. In fact, the best F1 score is 0.79 and is given by C1 with $\theta < 0.001$. However, both precision and recall drop for higher scores of θ. In contrast, C2 and C3 are quite stable and keep high precision values with a decent recall even for large divergence thresholds. The reason for this behavior could lay in the quality of the semantic annotations, and more exactly in the boundaries of the annotations. C1 gives the best clustering when considering only relations with very low divergence (i.e., relations which have a well-defined signature). However, when augmenting θ, many relation pairs that have large divergence are sneaking in and becoming part of the clusters. The fact that in C1 there are many relation pairs with large divergence can only be due to two reasons: either the relations are very general or there are errors in the annotation process. As general relations are discarded by the *tf-idf* filter, we believe the sensitivity of C1 to the divergence is due to errors in the annotations. As the annotation of C1 is the result of the harmonization of four systems, if the

boundaries of the annotation are not well detected, the annotation will have an erroneous semantic type assigned, which has a direct impact on the divergence of the relations w.r.t. the signatures. An overview of the generated clusters revealed this fact. The difference in the number and size of the clusters and coverage of C1 w.r.t. C2 and C3 is also noticeable, getting the latter more stable clusters for different thresholds.

7.3. Comparison of distributional similarity metrics

The similarity measure applied in our method can be categorized into the group of distributional similarity metrics as it calculates the similarity of two relation strings based on the arguments with which they occur. The innovative aspect is the use of semantic annotation to calculate the similarity based on entities (and not on strings) and with respect to the semantic types associated to the entities, which constitute the signature of a relation. In this experiment, we take as input the second configuration of the CALBC corpus (C2) (i.e., corpus annotated with CMA tool and LSP as pattern extractor) and compare the clustering output using four different similarity measures: *Lin*, *DIRT*, *Cover* and *BInc*. Table 6 shows the results.

The metrics are calculated for different thresholds of the similarity measures. For low similarity thresholds all the measures have a similar behavior, that is, even though the recall and coverage are quite high, the precision of the clusters is very low. If we increase the similarity threshold to gain precision, the coverage drops dramatically, except for *Cover*, which maintains an acceptable coverage but the recall is still low. By comparing the metrics obtained with these similarity measures with the ones proposed in this paper (C2 of Table 5), we observe that the F1 score is much higher in our method. This corroborates the benefits of using semantic annotation in distributional similarity measures.

7.4. Comparison with UMLS Semantic Network

The UMLS Semantic Network provides a broad categorization of the semantic types and also a handful set of relationships between them. In this experiment, we take the Semantic Network as reference an manually compare a subset of the discovered abstract semantic relations with those defined in the Semantic Network. We selected a random sample of 222 clusters produced with the configuration C2 and satisfying the property of having more than 50 classified relation instances. For each cluster, an expert assessed the correctness of the abstract semantic relation by comparing it with the relations of the Semantic Network. For the overlapping relations (i.e., relations having the same signature types), which amount to 177 (80%), the expert manually assessed the precision, which is 96%. We also detected that 67% of the overlapping relations discovered by our method are more specific than the relations in the Semantic Network. For example, for the relation *produces* between *Cell* and *Amino Acid Peptide or Protein* we were able to find more specific relations such as *stimulated*, *cultured*, *secretes*, *activated*, *coated* or *expresses*. The rest of the relations discovered by our method, which are 45 (20%), are not covered by the Semantic Network. The accuracy of these relations was manually assessed by an expert and is 82%. Some examples include *exposed(Mammal, Hazardous or Poisonous Substance)* or *detected(Virus,*

Sim. measure	Metrics	Threshold for the similarity measure				
		0.1	0.2	0.3	0.4	0.5
Lin	P	0.25	0.41	0.53	0.45	-
	R	0.71	0.62	0.58	0.86	-
	F1	0.30	0.40	0.45	0.55	-
	Clusters	919	471	112	17	-
	Avg. cluster size	11.32	5.21	4.06	3.65	-
	Coverage (%)	60.11	35.93	10.86	0.91	-
DIRT	P	0.24	0.30	0.33	0.53	0.79
	R	0.82	0.73	0.69	0.60	0.40
	F1	0.30	0.34	0.36	0.48	0.41
	Clusters	1179	782	276	42	8
	Avg. cluster size	36.51	18.97	10.23	4.76	3.75
	Coverage (%)	73.76	59.46	32.53	5.65	0.85
Cover	P	0.19	0.26	0.34	0.38	0.44
	R	0.80	0.73	0.69	0.65	0.64
	F1	0.27	0.32	0.37	0.40	0.44
	Clusters	995	834	484	332	155
	Avg. cluster size	20.28	11.91	8.48	6.80	5.89
	Coverage (%)	64.81	62.41	49.70	40.61	26.79
BInc	P	0.22	0.36	0.46	0.55	0.44
	R	0.76	0.65	0.59	0.57	0.78
	F1	0.28	0.38	0.42	0.47	0.56
	Clusters	990	646	248	71	12
	Avg. cluster size	15.40	7.22	5.05	3.61	3.17
	Coverage (%)	64.29	51.40	27.03	7.21	0.57

Table 6. Evaluation of the clustering method using different similarity measures.

Body Substance). The incorrect relations are mainly due to annotation errors. For example, the relation $eluted(Amino\ Acid\ Peptide\ or\ Protein, Finding)$ is judged as incorrect. By taking a closer look at the facts classified under this relation we discover facts such as $< antibodies, eluted\ from, glomeruli >$, where *antibodies* has been correctly annotated but *glomeruli* has been assigned the semantic type *Finding*.

7.5. Examples of clusters, synonyms and facts

In this section we show some interesting examples of both clusters, synonymous relations and facts that have been discovered by our method. Table 7 shows examples of synonymous relation strings associated to the corresponding semantic relation. Notice that only the head of the synonymous relation pattern is shown. Table 8 shows examples of relation instances extracted by the method for each of the clusters in Table 7.

#	Semantic Relation	Synonymous strings
1	cultured(Cell,Amino Acid)	cultivated,incubated,cultured
2	implicated(Amino Acid,Pathologic Function)	implied,contribute,implicated
3	infiltrating(Cell,Body Part)	entering,invading infiltrated,infiltrate,infiltrating
4	reconstituted(Mammal,Cell)	engrafted,grafted,repopulated transplanted,infused,reconstituted
5	analyzed(Body Substance,Laboratory Procedure)	analyzed,assayed,analysed
6	vaccinated(Mammal,Virus)	vaccinated,challenged,inoculated
7	develop(Mammal,Disease or Syndrome)	develop,developing,susceptible

Table 7. Examples of clusters of synonymous relation strings.

#	Knowledge Facts
1	< mouse [fibroblasts], were incubated with, [lymphokine] >
2	< [proteases], may contribute to the, pathogenesis of [chronic onchocercal dermatitis] >
3	< [lymphocytes], infiltrating the, [thyroid gland] >
4	< [SCID mice], were engrafted with, [peripheral blood lymphocytes] >
5	<[bone marrow samples] from patients with multiple myeloma, were analyzed by, [flow cytometry] >
6	< [monkeys], were vaccinated with, live-attenuated [SHIV] >
7	< [pigs], were highly susceptible to, [Salmonella infections] >

Table 8. Examples of extracted relation instances. Notice the head of the subject and object is in between brackets.

To further emphasize the variety and usefulness of semantic relations that the system is able to extract, we simply run a query to extract the relation instances talking about *asthma*. We obtain nearly 400 relation instances characterizing this disease, from which we show an excerpt in Table 9. The meaning for the semantic type codes is shown in Table 10.

8. Conclusions and Future Work

This paper addresses open IE in the biomedical domain from a semantic viewpoint in an unsupervised manner. We claim semantic annotation of scientific literature is the key to discover and extract groups of semantic relations so that they can augment the already available knowledge resources, as the attached semantics allows machine readability, inferencing and reusability of the extracted knowledge. The proposed synonymy model and clustering algorithm also benefit from the semantics of the annotations to group synonymous relation strings denoting new abstract relations with the appropriate signature types. The empirical evaluation shows that our method is efficient and can perform well on large

Knowledge Facts	**Abstract Relation**
asthma induced by *isocyanates*	induced(T047,T109)
asthma induced by *Candida antigen*	induced(T047,T121)
asthma induced by *influenza*	induced(T047,T047)
asthma induced by *ovalbumin*	induced(T047,T116)
asthma complicated by *fungal infections*	complicated(T047,T047)
asthma complicated by *asthma*	complicated(T047,T047)
asthma is characterized by *infiltration* of inflammatory cells	characterized(T047,T046)
asthma is characterized by *eosinophilia*	characterized(T047,T047)
asthma characterized by *airway hyperreactivity*	characterized(T047,T047)
asthma are characterized by *chronic inflammation*	characterized(T047,T046)
organic dust-induced [asthma] characterized by airway *neutrophilia*	characterized(T047,T033)
asthma accompanied by *rhinitis*	accompanied(T047,T047)
cytokines implicated in *asthma*	implicated(T116,T047)
outdoor *allergens* is strongly associated with *asthma*	strongly-positive(T129,T047)
dog *allergens* was strongly associated with *asthma*	strongly-positive(T129,T047)
parasite *infection* might prevent *asthma*	prevent(T047,T047)
allergen specific immunotherapy to treat *asthma*	prevent(T061,T047)
exercise-induced *asthma* treated with *anti-leukotrienes*	treated(T047,T109)
type I *asthma* mediated by *IgE*	mediated(T047,T116)

Table 9. An excerpt of the extracted relation instances characterizing asthma disease. Notice we use the semantic type codes of UMLS to specify the relation arguments for space restrictions.

Sem. type code	**Textual label**
T109	Organic Chemical
T116	Amino Acid Peptide or Protein
T121	Pharmacologic Substance
T129	Immunologic Factor
T033	Finding
T046	Pathologic Function
T047	Disease or Syndrome
T061	Therapeutic or Preventive Procedure

Table 10. Semantic type's textual label.

scale data. Moreover, the method is capable of extracting high quality semantic relations and groups of synonyms from unstructured text.

This research opens broad ways for future improvements and extensions. About the lessons learned from the experiments we emphasize the impact of the semantic types assigned to the annotated entities. In order for the method to perform well, we require minimum quality guarantees in the semantic annotation process. One future research line is the development of semantic annotation methods oriented to the task of relation extraction. We would also like to further investigate new techniques to capture non-frequent relations and couple them with our method in an attempt to improve recall while keeping the good precision

rates achieved. On the other hand, we are also studying how to use the extracted abstract relations for disambiguation purposes. That is, to apply some kind of bootstrapping to ambiguous annotations and extract more relations. Finally, it would be interesting to consider the learning of rules over the discovered relation predicates in order to uncover implicit relations [7, 44].

Acknowledgements. We thank anonymous reviewers for their very useful comments and suggestions. The work was supported by the CICYT project TIN2011-24147 from the Spanish Ministry of Economy and Competitiveness (MINECO).

References

[1] C. B. Ahlers, M. Fiszman, D. Demner-Fushman, F.-M. Lang, and T. C. Rindflesch. Extracting Semantic Predications from Medline Citations for Pharmacogenomics. In R. B. Altman, A. K. Dunker, L. Hunter, T. Murray, and T. E. Klein, editors, *Pacific Symposium on Biocomputing*, pages 209–220. World Scientific, 2007.

[2] A. Aronson. Effective mapping of Biomedical text to the UMLS metathesaurus: the MetaMap program. *Proc AMIA Symp*, pages 17–21, 2001.

[3] A. R. Aronson and F.-M. Lang. An overview of MetaMap: historical perspective and recent advances. *JAMIA*, 17(3):229–236, 2010. `http://metamap.nlm.nih.gov/`.

[4] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. DBpedia: a nucleus for a web of open data. In *Proceedings of the 6th international semantic web conference and 2nd Asian conference on Asian semantic web conference*, ISWC'07/ASWC'07, pages 722–735, Berlin, Heidelberg, 2007. Springer-Verlag. `http://dbpedia.org`.

[5] M. Banko, M. J. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni. Open Information Extraction from the Web. In M. M. Veloso, editor, *IJCAI*, pages 2670–2676, 2007.

[6] M. Banko and O. Etzioni. The Tradeoffs Between Open and Traditional Relation Extraction. In *ACL*, pages 28–36. The Association for Computer Linguistics, 2008.

[7] J. Berant, I. Dagan, and J. Goldberger. Global Learning of Typed Entailment Rules. In D. Lin, Y. Matsumoto, and R. Mihalcea, editors, *ACL*, pages 610–619. The Association for Computer Linguistics, 2011.

[8] R. Berlanga, V. Nebot, and E. Jimnez-Ruiz. Semantic annotation of biomedical texts through concept retrieval. *Procesamiento de Lenguaje Natural*, 45:247–250, 2010.

[9] T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, May 2001.

[10] O. Bodenreider. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(Database-Issue):267–270, 2004. `http://www.nlm.nih.gov/research/umls`.

[11] D. T. Bollegala, Y. Matsuo, and M. Ishizuka. Relational duality: unsupervised extraction of semantic relations between entities on the web. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 151–160, New York, NY, USA, 2010. ACM.

[12] M. Bundschus, M. Dejori, M. Stetter, V. Tresp, and H.-P. Kriegel. Extraction of semantic biomedical relations from text using conditional random fields. *BMC Bioinformatics*, 9, 2008.

[13] K. B. Cohen, M. Palmer, and L. Hunter. Nominalization and Alternations in Biomedical Language. *PLoS ONE*, 3(9):e3158, 09 2008.

[14] A. Coulet, N. H. Shah, Y. Garten, M. Musen, and R. B. Altman. Using text to build semantic networks for pharmacogenomics. *J. of Biomedical Informatics*, 43:1009–1019, December 2010.

[15] D. Rebholz-Schuhmann et al. CALBC silver standard corpus. *J Bioinform Comput Biol*, 8(1):163–79, 2010.

[16] M. Dai, N. Shah, W. Xuan, M. Musen, S. Watson, B. Athey, and F. Meng. An efficient solution for mapping free text to ontology terms. In *American Medical Informatics Association Symposium on Translational BioInformatics*, AMIA-TBI'08, 2008.

[17] C. de Pablo-Sánchez, I. Segura-Bedmar, P. Martínez, and A. Iglesias-Maqueda. Lightly supervised acquisition of named entities and linguistic patterns for multilingual text mining. *Knowledge and Information Systems*, pages 1–23. 10.1007/s10115-012-0502-0.

[18]A. Fader, S. Soderland, and O. Etzioni. Identifying relations for open information extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1535–1545, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[19]C. Giuliano, A. Lavelli, and L. Romano. Exploiting Shallow Linguistic Information for Relation Extraction from Biomedical Literature. In *EACL*. The Association for Computer Linguistics, 2006.

[20]Z. Harris. Distributional structure. *Word*, 10(23):146–162, 1954.

[21]L. Hirschman, M. E. Colosimo, A. A. Morgan, and A. S. Yeh. Overview of biocreative task 1b: normalized gene lists. *BMC Bioinformatics*, 6(S-1), 2005.

[22]M. Huang, X. Zhu, S. Ding, H. Yu, and M. Li. ONBRIRES: Ontology-Based Biological Relation Extraction System. In T. Jiang, U.-C. Yang, Y.-P. P. Chen, and L. Wong, editors, *APBC*, pages 327–336. Imperial College Press, London, 2006.

[23]M. Huang, X. Zhu, and M. Li. A hybrid method for relation extraction from biomedical literature. *I. J. Medical Informatics*, 75(6):443–455, 2006.

[24]T. K. Jenssen, A. Laegreid, J. Komorowski, and E. Hovig. A literature network of human genes for high-throughput analysis of gene expression. *Nature genetics*, 28(1):21–28, May 2001.

[25]J.-H. Kim, A. Mitchell, T. K. Attwood, and M. Hilario. Learning to extract relations for protein annotation. In *ISMB/ECCB (Supplement of Bioinformatics)*, pages 256–263, 2007.

[26]A. Kiryakov, B. Popov, I. Terziev, D. Manov, and D. Ognyanoff. Semantic annotation, indexing, and retrieval. *Web Semant.*, 2:49–79, December 2004.

[27]J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 111–119, New York, NY, USA, 2001. ACM.

[28]L. Lee. Measures of distributional similarity. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pages 25–32, Stroudsburg, PA, USA, 1999. Association for Computational Linguistics.

[29]J. Li, Z. Zhang, X. Li, and H. Chen. Kernel-based learning for biomedical relation extraction. *J. Am. Soc. Inf. Sci. Technol.*, 59:756–769, March 2008.

[30]D. Lin. Automatic retrieval and clustering of similar words. In *Proceedings of the 17th international conference on Computational linguistics - Volume 2*, COLING '98, pages 768–774, Stroudsburg, PA, USA, 1998. Association for Computational Linguistics.

[31]D. Lin and P. Pantel. Discovery of inference rules for question-answering. *Nat. Lang. Eng.*, 7(4):343–360, Dec. 2001.

[32]T. McIntosh, L. Yencken, J. R. Curran, and T. Baldwin. Relation Guided Bootstrapping of Semantic Lexicons. In *ACL (Short Papers)*, pages 266–270. The Association for Computer Linguistics, 2011.

[33]V. Nebot, M. Ye, J.-H. Eom, and G. Weikum. DIDO: a DIsease-Determinants Ontology from web sources. In *WWW (Companion Volume)*, pages 237–240. ACM, 2011.

[34]S. Nelson, S. J. Nelson, A. R. Aronson, T. E. Doszkocs, and H. F. C. Ms. Automated assignment of medical subject headings. In *Proceedings of the American Medical Informatics Association (AMIA) Annual Symposium*, ACL '04, 1999. `http://www.nlm.nih.gov/mesh/`.

[35]A. Névéol and Z. Lu. Automatic integration of drug indications from multiple health resources. In *Proceedings of the 1st ACM International Health Informatics Symposium*, IHI '10, page 666673, New York, NY, USA, 2010. ACM. ACM ID: 1883096.

[36]J. C. Park, H. S. Kim, and J. jae Kim. Bidirectional Incremental Parsing for Automatic Pathway Identification with Combinatory Categorial Grammar. In *Pacific Symposium on Biocomputing*, pages 396–407, 2001.

[37]M. Pérez-Catalán, R. Berlanga, I. Sanz, and M. Aramburu. A semantic approach for the requirement-driven discovery of web resources in the Life Sciences. *Knowledge and Information Systems*, pages 1–20. 10.1007/s10115-012-0498-5.

[38]D. Rebholz-Schuhmann, M. Arregui, S. Gaudan, H. Kirsch, and A. Jimeno-Yepes. Text processing through Web services: calling Whatizit. *Bioinformatics*, 24(2):296–298, 2008. `http://www.ebi.ac.uk/webservices/whatizit/info.jsf`.

[39]D. Rebholz-Schuhmann, A. Jimeno-Yepes, M. Arregui, and H. Kirsch. Measuring prediction capacity of individual verbs for the identification of protein interactions. *Journal of Biomedical Informatics*, 43(2):200–207, 2010.

[40]L. Reeve and H. Han. Survey of semantic annotation platforms. In *Proceedings of the 2005*

ACM symposium on Applied computing, SAC '05, pages 1634–1638, New York, NY, USA, 2005. ACM.

[41] L. H. Reeve and H. Han. CONANN: an online biomedical concept annotator. In *Proceedings of the 4th international conference on Data integration in the life sciences*, DILS'07, pages 264–279, Berlin, Heidelberg, 2007. Springer-Verlag.

[42] E. Riloff and R. Jones. Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping. In J. Hendler and D. Subramanian, editors, *AAAI/IAAI*, pages 474–479. AAAI Press / The MIT Press, 1999.

[43] B. Rosario and M. A. Hearst. Classifying semantic relations in bioscience texts. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, Stroudsburg, PA, USA, 2004. Association for Computational Linguistics.

[44] S. Schoenmackers, O. Etzioni, D. S. Weld, and J. Davis. Learning first-order Horn clauses from web text. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 1088–1098, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[45] Y. Shinyama and S. Sekine. Preemptive information extraction using unrestricted relation discovery. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 304–311, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[46] K. Spackman. SNOMED RT and SNOMED CT. promise of an international clinical terminology. *MD Comput*, 17(6):29, 2000. `http://www.nlm.nih.gov/research/umls/Snomed/snomed_main.html`.

[47] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA, 2007. ACM. `http://www.mpi-inf.mpg.de/yago-naga/yago/`.

[48] F. M. Suchanek, M. Sozio, and G. Weikum. SOFIE: a self-organizing framework for information extraction. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 631–640, New York, NY, USA, 2009. ACM.

[49] I. Szpektor and I. Dagan. Learning entailment rules for unary templates. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 849–856, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[50] L. Tari, S. Anwar, S. Liang, J. Cai, and C. Baral. Discovering drug-drug interactions: a text-mining and reasoning approach based on properties of drug metabolism. *Bioinformatics*, 26(18), 2010.

[51] J. M. Temkin and M. R. Gilder. Extraction of protein interaction information from unstructured text using a context-free grammar. *Bioinformatics*, 19(16):2046–2053, 2003.

[52] J. Weeds and D. Weir. A general framework for distributional similarity. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, pages 81–88, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[53] A. Yates, M. Cafarella, M. Banko, O. Etzioni, M. Broadhead, and S. Soderland. TextRunner: open information extraction on the web. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, NAACL-Demonstrations '07, pages 25–26, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

[54] A. Yates and O. Etzioni. Unsupervised Methods for Determining Object and Relation Synonyms on the Web. *J. Artif. Intell. Res. (JAIR)*, 34:255–296, 2009.

[55] G. Zhou, D. Shen, J. Zhang, J. Su, and S.-H. Tan. Recognition of protein/gene names from text using an ensemble of classifiers. *BMC Bioinformatics*, 6(S-1), 2005.

[56] J. Zhu, Z. Nie, X. Liu, B. Zhang, and J.-R. Wen. StatSnowball: a statistical approach to extracting entity relationships. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 101–110, New York, NY, USA, 2009. ACM.

Author Biographies

Victoria Nebot received her B.S. degree in Computer Science from Universitat Jaume I, Spain, in 2007. She joined the Temporal Knowledge Bases Group (TKBG) at Universitat Jaume I as a PhD Student in 2008. Her main research is focused on analyzing and exploiting semi-structured and complex data derived mainly from the Semantic Web. Also, she is interested in text mining and information extraction methods that contribute to the growing of the Semantic Web.

Rafael Berlanga is full professor of Computer Science at Universitat Jaume I, Spain, and the leader of the TKBG research group. He received the B.S. degree from Universidad de Valencia in Physics, and the PhD degree in Computer Science in 1996 from the same university. His current research interests include text mining, knowledge bases, information retrieval, and the semantic web. He has led several research projects, has published more than 20 contributions to high-impact international journals and more than 50 contributions to international conferences.

Correspondence and offprint requests to: Victoria Nebot, Departamento de Lenguajes y Sistemas Informaticos, Universitat Jaume I, Castellón, 12071, Spain. Email: romerom@uji.es