Tailored Semantic Annotation for Semantic Search

Rafael Berlanga, Victoria Nebot*, María Pérez

Departament of Computer Languages and Systems,
Universitat Jaume I, Avda. Vicent Sos Baynat s/n, 12071 Castellón, Spain

Abstract

This paper presents a novel method for semantic annotation and search of a target corpus using several knowledge resources (KRs). This method relies on a formal statistical framework in which KR concepts and corpus documents are homogeneously represented using statistical language models. Under this framework, we can perform all the necessary operations for an efficient and effective semantic annotation of the corpus. Firstly, we propose a coarse tailoring of the KRs w.r.t the target corpus with the main goal of reducing the ambiguity of the annotations and their computational overhead. Then, we propose the generation of concept profiles, which allow measuring the semantic overlap of the KRs as well as performing a finer tailoring of them. Finally, we propose how to semantically represent documents and queries in terms of the KRs concepts and the statistical framework to perform semantic search. Experiments have been carried out with a corpus about web resources which includes several Life Sciences catalogues and Wikipedia pages related to web resources in general (e.g., databases, tools, services, etc). Results demonstrate that the proposed method is more effective and efficient than state-of-the-art methods relying on either context-free annotation or keyword-based search.

Keywords: Semantic Annotation, Semantic Search, Language Models.

1. Introduction

Semantic annotation is the process of linking the meaning of unstructured data to concepts that are unambiguously described in a knowledge resource (KR). Automatic semantic annotation is playing a crucial role in a great variety of applications of the Semantic Web such as linked data generation, open information extraction, ontology alignment, and semantic search. Specifically, semantic search allows users to express their information needs in terms of concepts taken from one or several KRs. Unlike traditional keyword-based searches, semantic search can make use of the KR semantic relationships to perform new tasks such as to refine the user queries with broader or more specific concepts of the KR, to browse the whole content of the collection through the taxonomies provided by the KRs, and to provide friendlier visualizations to explore the retrieved documents [6]. Successful applications like PubMed/Medline [1], the most popular search engine for the biomedical community, have demonstrated the enormous potential that semantic annotations have for end-users and third-party information consumer applications. Unfortunately, PubMed/Medline relies on manual semantic indexing performed by experts, which cannot be extrapolated to other domains and other scenarios that require massive annotation of texts such as opinion analysis. As a consequence, there is currently a great demand of fully automatic annotation methods.

Automatic semantic annotation has been widely

applied to Life Sciences. For example, the biomedical community is interested in finding out new relationships between biological systems and clinical research. Many text mining approaches rely on the semantic annotation of the scientific literature in order to identify relevant biomedical entities such as proteins, genes, diseases, etc. and their relationships [41]. Outside the Life Sciences area, semantic annotation has been mainly focused on Named Entities such as people, organizations, places, etc.[1]. Most of these methods rely on the dictionary look-up approach, which consists in identifying the entities mentioned in a text by looking for slightly variants of them in the KR lexicon. It is well known that for open target collections and large KRs, a text chunk can match several concepts of the KR, leading this way to the ambiguity issue. Even in specialized scenarios like Biomedicine, ambiguity can produce noise enough to hamper the effectiveness of the semantic searches (this will be further discussed in Section 4).

Several decades of research on word sense disambiguation (WSD) have demonstrated how hard it is to deal with ambiguity in natural language processing. Traditionally, WSD has been defined in terms of an inventory of word senses (e.g., WordNet). A WSD method aims at selecting the right senses for the words present in a text. Two main trends have been explored in the literature [39] namely, supervised approaches, which learn how to disambiguate a word given a series of examples about its senses, and knowledge-based approaches, which use the KR information to select the right sense without supervision. Nowadays the application of existing WSD methods to automatic semantic annotation is an open challenge due to three main issues: (1) semantic annotation must deal with arbitrary and usually large KRs, (2) the WSD method must be highly scalable in order to annotate very large collections, and (3) they should deal with the incompleteness of a KR, which usually does not contain all the possible senses of a term. Currently, the first issue makes supervised methods impractical, as we cannot gather examples of use for all the concepts in the KRs. The second issue makes current knowledge-based methods very time consuming as they need either to compare very large profiles of terms (e.g., [2, 22, 4]) or to compute large graphs with all senses involved at each sentence (e.g., [40, 16]). As for the third issue, most WSD methods will consider unambiguous many strings since only one sense is covered in the KR.

In this paper we propose a novel method to perform context-based semantic annotation. The goal of this method is not only to find the concepts that best lexically fit in with the target text but also that their latent meanings fit in as well with those of the corpus to be annotated. We evaluate our semantic annotation method for semantic search tasks, in particular, for web resource discovery, where user queries are heterogeneous and usually expressed in a high level of abstraction.

The outline of the paper is as follows: in Section 2 we state the main contributions and novelties of the proposed method. Section 3 introduces some notation and background about the underlying foundations and Section 4 discusses the main semantic annotation issues overcome by our method. In Section 5 we present the proposed method and then explain each of the components. Section 6 is devoted to the experimental evaluation. In Section 7 we review current related work and in Section 8 we discuss the main conclusions and future work.

2. Contribution

The main contribution of the paper is a novel method for performing context-based semantic annotation and search based on a statistical formal background, more specifically, on statistical language models. The main novelties of this method are:

- It is able to deal with several arbitrary and large KRs.
- Unlike current Wikipedia and UMLS®-based annotators, our method is independent of the specific characteristics of each KR. For example, it does not make use of disambiguation pages, internal links and other Wikipedia specific features.

[1] Text Analysis Conference: `http://www.nist.gov/tac/2013/KBP/`

- It is able to deal with both global and local contexts in order to validate the generated annotations.
- It uses a statistical framework for performing all the required operations for semantic indexing and search.

As far as we know, this is the first time language models are used to define a formal framework for semantic annotation and search. Statistical language models have provided in the last decades a sound background to perform most of text processing tasks, such as information retrieval, text categorization and automatic text translation. Language models define a theoretical framework to represent and operate over text semantics in terms of word distributions. The main advantage of these models is that they do not require any kind of natural language processing, making them quite attractive to define scalable methods for automatic semantic annotation and semantic search. Moreover, in this paper we show how language models can be naturally used to tailor KRs to target corpus in order to reduce ambiguity and increase efficiency.

3. Background

In this section we introduce the concepts and foundations that underlie the developed method. First, we define the concept of KR. Then, we define the notion of semantic annotation. Finally, we introduce statistical language models as the main foundation of our approach.

3.1. Knowledge Resource

In the following, we formalize the concept of KR and the minimal elements it must provide in order to be useful for semantic annotation and search.

Definition 3.1. *A knowledge resource is a formalization of the semantics of a domain by means of a set of concepts* $\mathcal{C} = \{c_1, ..., c_n\}$*. A concept* $c \in \mathcal{C}$ *represents the semantic definition of a meaningful entity in a specific domain.*

In order to find out candidate concepts for a text chunk, the KR must provide a lexicon describing its concepts. We assume that there exists a function $lex(c)$ that returns the set of strings describing the concept c (e.g., labels, synonyms, etc). This set of strings can contain different lexical variants of c and synonyms of these variants. Moreover, we also assume that the KR provides a function $def(c)$ that provides a short description of the concept.

The concepts in a KR can be taxonomically related by their subsumption (is-a) or by "broader-than" relationships. The taxonomic relationship between two concepts c and c' is represented as $c \preceq c'$. A KR can provide other concept relationships but they are not considered in our approach.

In this work, we make use of the largest and most popular KRs currently used for semantic search: UMLS® and Wikipedia. For the latter, we adopt Wikinet [38] since it fits to our definition of KR.

For illustration purposes, we show an example of the information that Wikinet provides for the concept with identifier "W11258494":

```
lex(c)= { residue, chemical residue }
def(c)= ‘‘In chemistry, residue is the material
remaining after a distillation or an evaporation of
a methyl group.  It may also refer to the undesired
byproducts of a reaction’’
```

3.2. Semantic Annotation

Performing the semantic annotation of a document d consists in finding mappings between text chunks t of d (i.e., sequences of adjacent terms), and the concepts that best semantically describe the contents of d. As concepts of a KR are usually expressed as noun phrases, text chunks are usually associated to these syntactic structures. We formally define semantic annotation as follows:

Definition 3.2. *Given a knowledge resource KR, and* $d = (w_1, ..., w_n)$ *a document (i.e., input set of sequences over terms from the vocabulary* $\mathcal{V}$*), a semantic annotation is a pair* $< c, t >$ *where* $c \in KR$ *and* t *is a subsequence of* d *such that there exists a mapping from lex(c) to a subset t',* $t' \subseteq t$*.*

Next, we show an example of semantic annotation of the sentence: *BriX is a database containing some protein fragments from 4 to 14 residue from protein homology*, which is a description of a database. Each annotation includes the KR name (i.e., *src*), the concept identifier (i.e., *cui*), the semantic type and semantic group (i.e., *type* and *grp* resp.), the offset and length of the annotation in the text (i.e., *offset* and *len*).

```
<e id="doc1.e1" src="WIKINET" cui="W001369226" offset="0"
len="4">Brix</e>
<e id="doc1.e2" src="UMLS" cui="C1335533" type="T116"
grp="CHEM" offset="40" len="17">protein fragments</e>
<e id="doc1.e3" src="WIKINET" cui="W014134516"
offset="40">protein</e>
<e id="doc1.e4" src="UMLS" cui="C1709915" type="T077"
grp="CONC" offset="71" len="7">residue</e>
<e id="doc1.e5" src="UMLS" cui="C1334043" type="T028"
grp="GENE" offset="92" len="8">homology</e>
<e id="doc1.e6" src="UMLS" cui="C2697616" type="T080"
grp="CONC" offset="92" len="8">homology</e>
<e id="doc1.e7" src="UMLS" cui="C0162775" type="T081"
grp="CONC" offset="92" len="8">homology</e>
<e id="doc1.e8" src="WIKINET" cui="W010746546"
offset="84" len="16">protein homology</e>
```

Definition 3.3. *Given a KR, a document d and its set of semantic annotations E_d, a semantic annotation $< c, t >$ is ambiguous if there exists another semantic annotation $< c', t' > \in E_d$ where $c \neq c'$ and $t = t'$.*

A semantic annotation is ambiguous if more than one concept has been assigned the exact same subset of tokens. In the previous example, the string *homology* has been annotated with three different concepts from UMLS® that belong to different semantic types (i.e., Quantitative concept, Qualitative concept and Gene or Genome).

Current automatic annotation is performed independently from the context in which concepts are identified, assuming that the lexicons are well suited to the corpus to be annotated. However, the semantics of a concept may not fit in with the context in which it occurs. Additionally, we have the problem of erroneously assigning a unique concept to a text chunk because the correct concept is not present in the KR. To detect these cases we also need to take into account the context of the generated annotations. Next, we present the main foundation used to validate semantic annotations, which uses statistical language models to characterize both the KR concepts and the context where the annotations take place.

3.3. Statistical Language Models

In order to characterize the KRs and the corpora to be annotated, as well as to capture the main contexts they can generate, we adopt a statistical framework based on language models. A statistical language model assigns a probability to a sequence of n words $p(w_1, \ldots, w_n)$ by means of a probability distribution.

Let $\mathcal{V} = \{w_1, \cdots, w_N\}$ be the vocabulary used in the KRs as well as the corpora to be annotated. We consider that any text description d consists of an observed sequence of terms $(w_1, \cdots, w_k)$ with $w_i \in \mathcal{V}$ for which a language model θ_d can be associated. This language model represents the word distribution $\{p(w|\theta_d)\}_{w \in \mathcal{V}}$. When this distribution is estimated via Maximum Likelihood Expectation (MLE), we denote the model as $\widehat{\theta}_d$. MLE only uses the relative frequency of the terms in d (i.e., $p(w|d) \propto tf(w, d)$). Due to the sparsity of $\widehat{\theta}_d$, several smoothing approaches have been proposed to estimate more appropriate models for d (e.g., Dirichlet prior and Jelinek-Mercer). Basically, the goal with these techniques is to build an approximate model $\tilde{\theta}_d$ by using the global information provided by a background corpus over the same vocabulary $\mathcal{V}$. As our aim is to validate the annotations by characterizing the KR concepts assigned (i.e., building richer concept profiles) and capturing the context where the annotation occurs, we will focus on smoothing techniques based on statistical translation [28].

A translation model estimates the translation probabilities between the words of a given corpus $\mathcal{G}$. We represent this translation model as $T_{\mathcal{G}} = \{p(w|w')\}_{w,w' \in \mathcal{V}}$ where $p(w|w')$ indicates the probability of observing w if we have observed w' in a given context. Statistical translation has been used in information retrieval (IR) for query expansion [21] and

recommendation systems [45]. In our paper, translation models are mainly used to get richer profiles for the KR concepts.

The MLE estimation of a translation model $T_{\mathcal{G}}$, denoted $\widehat{T}_{\mathcal{G}}$, can be performed by applying the following formulas:

$$p(w|w') = \frac{p(w,w')}{p(w')} \quad (1)$$

$$p(w|w') \propto \sum_{s \in \mathcal{W}} p(w|s)p(w'|s)p(s) \quad (2)$$

$$p(w) = \sum_{w' \in \mathcal{V}} p(w,w') \quad (3)$$

This estimation requires a set of local contexts $\mathcal{W}$ taken from the target corpus $\mathcal{G}$, in which word co-occurrence is estimated. In this work, we define these local contexts by moving a window of fixed size across the whole collection [18]. In this case, $p(s) = 1/|s|$ and $p(w|s)$ is estimated by counting the occurrences of w in the context s.

The computation of translation models can be efficiently performed when the size of local contexts are relatively small (around 4-6 words). Moreover, the implementation of this computation can be massively distributed and parallelized[31].

Several techniques have been proposed to smooth translation models, all of them relying on random walks techniques. Thus, a k-step random walk of $\widehat{T}_{\mathcal{G}}$ with diffusion factor α can be calculated as follows:

$$\widetilde{T}_{\mathcal{C}} = (1-\alpha) \cdot \alpha^k \cdot \widehat{T}_{\mathcal{G}}^k \quad (4)$$

When $k \to \infty$ we obtain the eigen-based smoothing of $\widehat{T}_{\mathcal{G}}$, which has been widely adopted for document classification and spectral clustering [45]. In this paper, we will use these kernels only for smoothing semantic query models across concept taxonomies. For computational reasons, translation models of the corpora and the KRs will be smoothed with a 1-step random walk (i.e., k=1).

There exist alternative ways to refine the language models associated to documents and queries. In this paper we will use an adaptation of the parsimonious methods used in IR [21]. Basically, these methods assume that the observed model for documents $\widehat{\theta}_d$ (res. queries) is a mixture of a document-specific model (θ_d^s) and a background model ($\theta_{\mathcal{B}}$). To determine the specific model, an Expectation Maximization algorithm [14] is applied in order to maximize the likelihood w.r.t. the observed model, that is:

$$-\sum_{i=1}^{n} P(w_i|\widehat{\theta}_d) \cdot log(\lambda P(w_i|\tilde{\theta}_d) + (1-\lambda) \cdot P(w_i|\theta_{\mathcal{B}}) \quad (5)$$

E-step:

$$Z_{w_i} = \frac{\lambda_{k-1} \cdot P_{k-1}(w_i|\theta_d^s)}{\lambda_{k-1} \cdot P_{k-1}(w_i|\theta_d^s) + (1-\lambda_{k-1}) \cdot P(w_i|\theta_{\mathcal{B}}))} \quad (6)$$

M-step:

$$\lambda_k = \sum_{i=1}^{n} P(w_i|\widehat{\theta}_d) \cdot Z_{w_i} \quad (7)$$

$$P_k(w_i|\theta_d^s) = P(w_i|\widehat{\theta}_d) \cdot Z_{w_i} \cdot \lambda_k^{-1} \quad (8)$$

Language models obtained with parsimonious smoothing play a similar role to the application of the inverse document frequency (IDF) in vector space models: meaningless terms will present higher probabilities in $\mathcal{B}$ and lower probabilities in θ_d^s.

All the language models defined over the vocabulary $\mathcal{V}$ fall in a $(|\mathcal{V}|-1)$-simplex space, which can be used to measure the distance between them. Thus, we can measure the distance between the models of KR concepts, corpus documents and queries. For this purpose, in this paper we adopt the Fisher geodesic distance [27], which is defined as follows:

$$D(\theta_A, \theta_B) = 2 \cdot arcos(\sqrt{\theta_A \cdot \theta_B}) \quad (9)$$

4. Semantic annotation issues

The main issue to be addressed when performing the semantic annotation of a document is the treatment of ambiguous, spurious and wrong annotations, especially when performing context-free semantic annotation.

An ambiguous annotation arises when a sequence of words in a text is assigned to more than one concept from the KR. There are two main factors

that characterize ambiguous annotations: the size of the matched text, and the specificity of the terms involved in the annotation. The latter factor can be measured with the inverse document frequency (IDF). Figure 1 shows the percentage of ambiguous annotations w.r.t. the number of words they comprise for the two evaluated KRs (UMLS® and Wikinet). Figure 2 shows the percentage of ambiguous annotations of one word w.r.t the word IDF also for UMLS® and Wikinet. As expected, most ambiguous annotations fall in the short-size and low-IDF regions. This fact has a great impact in the semantic annotation process as ambiguous annotations occur very frequently, producing considerable noise in the resulting annotated collection. Moreover, any WSD method will considerably overload the annotation process.

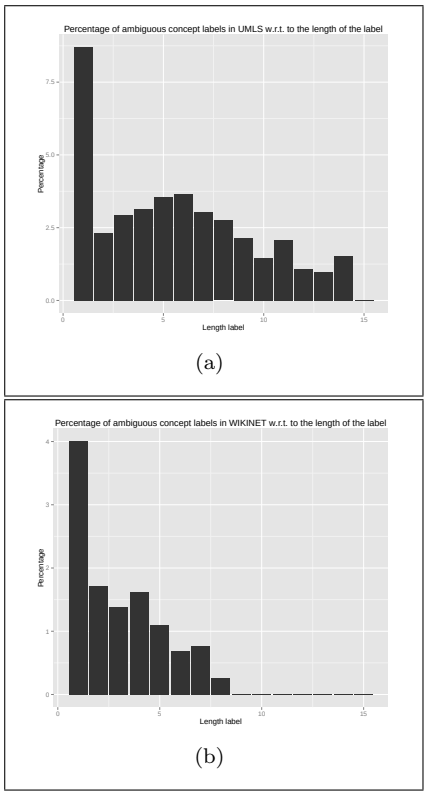

Figure 1: Ambiguity w.r.t. the length of the matched text.

Wrong annotations are those that involve a concept

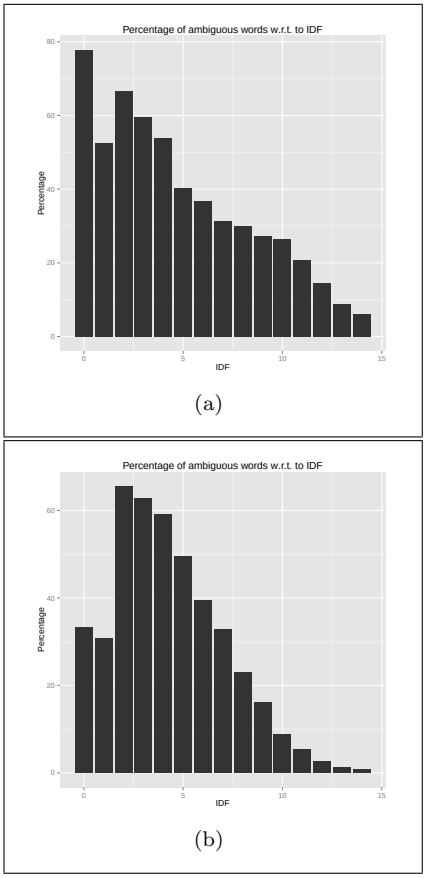

Figure 2: Ambiguity w.r.t. the IDF

whose meaning does not fit in at all with the context in which it is identified. These annotations are very frequent in acronyms and named entities such as programs, databases, algorithms, tools, and so on. Notice that WSD methods cannot reject wrong annotations since they are devised to choose at least one of the possible senses assigned to a word. However, the right sense of the word could be not included in the KR, and consequently it could be non-ambiguous for the KR.

Finally, spurious annotations are those that do not provide any value for performing semantic searches. Notice that the KRs have not been devised for semantic annotation but for representing knowledge. The KR can contain concepts that have only sense within

the KR, as they are used to organize and classify concept descriptions. These annotations also overload the semantic annotation process apart from introducing more noise to the annotated collection.

The method for semantic annotation and search that we present in the following section is aimed at reducing as much as possible the number of ambiguous, spurious and wrong annotations.

5. Method

In this section we present our method for semantic annotation and search, which is based on a fine tailoring of the KR based on the target corpus statistics. Moreover, we also propose to validate the generated annotations with the tailored KR by taking into account the contexts where they occur. As mentioned in the introduction, our hypothesis is that the better the tailoring process is, the less overhead and the more effectiveness we obtain in the semantic annotation process, thus reducing the number of ambiguous, spurious and wrong annotations.

Figure 5 sketches the proposed method. Starting from the original KRs and the corpus to be annotated, the first step consists in tailoring the KRs according to the corpus contents (step 1). This step is optional and aims to get coarse refinements of very large and heterogeneous KRs like Wikipedia. From the tailored (or original) KRs we estimate the language models for their concepts, that is, the concept profiles (step 2). These profiles can be similarly tailored using the corpus (step 3). The profiles can also be used to align concepts with similar lexica by assessing their contents overlap (step 4). Alignments can give us information about how much complementary they are as well as to reduce their redundancies. Once the concept profiles and their alignments are calculated, the semantic annotation of the corpus can be performed (step 5). The annotated corpus is then used to build the semantic document models (i.e., expressed in terms of the KRs concepts) which will be the basis for performing semantic searches. Queries are built by users by picking up concepts of interest from the tailored KRs (step 6). From these sets of concepts, an expanded query model is generated. Finally, document models are ranked according to their distance to the expanded query model, and presented to the user (step 6). In the following sections, we explain each of the main components in detail.

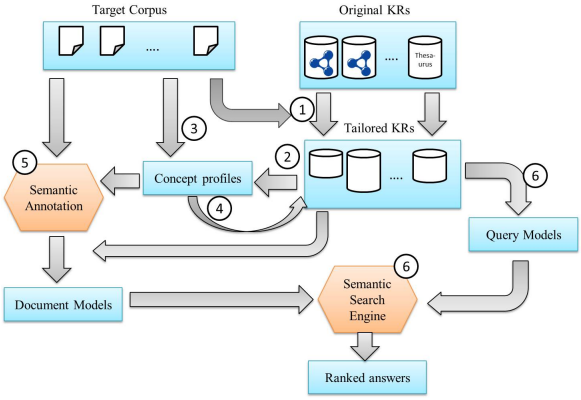

Figure 3: Summary of the proposed method for semantic annotation and search. The phases are: 1) tailoring of the KRs, 2) concept profile construction, 3) tailoring of concept profiles, 4) semantic overlap of the KRs, 5) context-based semantic annotation and 6) semantic search

5.1. Tailoring of a KR

We aim at selecting those concepts in the KR that are semantically related to the target corpus $\mathcal{G}$. For this purpose, we first calculate the unigram model of the KR lexicon ($\widehat{\theta}_{KR}$), which considers the texts returned by the functions $lex(c)$ and $def(c)$. This model is then refined by applying the EM procedure of Section 3.3 taking as background the corpus model $\widehat{\theta}_{\mathcal{G}}$. Let θ^s_{KR} be the resulting refined model. Then, each concept $c \in \mathcal{C}_{KR}$ is selected if its definition is more likely to be generated from θ_G than θ^s_{KR}, that is if $p(def(c)|\theta_G) > p(def(c)|\theta^s_{KR})$. Assuming the independence of the terms, $p(def(c)|\theta_.)$ can be estimated as:

$$p(def(c)|\theta_.) = \prod_{w \in def(c)} p(w|\theta_.) \qquad (10)$$

This test is aimed at filtering out those KR concepts that are completely out of context w.r.t. the target corpus. In very large and heterogeneous KRs

like Wikinet this coarse tailoring allows the system to manage a much smaller KR to efficiently perform semantic annotation.

5.2. Concept profile construction

For each concept in a KR, we build a concept profile based on language models as follows:

$$p(w|\theta_c) = \alpha \cdot p(w|\theta^{'}_{lex(c)}) + (1-\alpha) \cdot p(w|\theta^{'}_{def(c)}) \quad (11)$$

The model of the concept profile is based on a mixture of the models $\theta^{'}_{lex(c)}$ and $\theta^{'}_{def(c)}$ obtained from the lexical variants of the concept and the concept definition, respectively. They are calculated as follows:

$$p(w|\theta^{'}_f)_{f\in\{lex(c),def(c)\}} = \beta \cdot p(w|\widehat{\theta}_f) + (1-\beta) \cdot p(w|\theta^*_f) \quad (12)$$

$$p(w|\theta^*_f) = \sum_{w' \in \mathcal{V}} T_{KR}(w'|w) P(w'|\widehat{\theta}_c) \quad (13)$$

These models are at the same time a mixture of the MLE model and a smoothed model obtained by applying the translation model generated from the KR concept definitions (i.e., T_{KR}) to the MLE model. In this case, we apply a 1-step random walk as shown in formula 4.

The parameter α weights the contribution of the lexical variants vs. the definition of the concept and the parameter β weights the contribution of smoothed model generated by applying the translation model.

Moreover, we have also devised an extended version for the context's profiles based on the direct parents and children of the concept:

$$p(w|\theta^{ext}_c) = \gamma \cdot p(w|\theta_c) + (1-\gamma) \cdot \sum_{c' \preceq c, c \preceq c'} p(w|\theta_{c'}) p(\theta_{c'}) \quad (14)$$

The parameter γ serves to calibrate the contributions of the models of the parents and children. The prior $p(\theta_{c'})$ is assumed to be uniform. Parameters α, β and γ will be empirically set.

For example, the term "residue" has several concepts associated in Wikinet, and for each of them we generate a profile that differentiates their semantics. As a result, the concept that refers to the statistical residue contains as profile words such as *error*, *statistics* or *deviation*, the biochemistry residue profile contains *chemical*, *enzyme* or *protein*, and the residue referring to the taker of the residuary estate contains *wills*, *property* or *estate*. Next, we show an excerpt of the concept profiles for these three different concepts:

```
W000461509:  error:0.0864 sample:0.0562 statistics:0.0353
residual:0.0283 function:0.0263 deviation:0.0240
classical:0.0179 analysis:0.0124 optimization:0.0123
model:0.0121 measures:0.0120 square:0.0115 mean:0.0114
theoretical:0.0113 random:0.0111
W011258494:  residue:0.1775 chemical:0.0942
reaction:0.0425 group:0.0242 enzyme:0.0240
evaporation:0.0217 distillation:0.0211 chemistry:0.0207
material:0.0191 molecule:0.0189 catalysis:0.0189
protein:0.0089
W002266690:  residuary:0.1555 estate:0.0871 clause:0.0621
residue:0.0292 residual:0.0255 bequest:0.0255
legatee:0.0255 taker:0.0178 real:0.0167 property:0.0126
wills:0.0107 testator:0.01 male:0.0096 part:0.0090
```

5.3. Tailoring concept profiles

Given a concept c and its profile θ_c , we can measure the distance of the concept's profile w.r.t. the corpus as $D(\theta_c, \theta_{c,\mathcal{G}})$, where $\theta_{c,\mathcal{G}}$ is the joint distribution of θ_c and the translation model of the corpus:

$$p(w|\theta_{c,\mathcal{G}}) = \sum_{w' \in \mathcal{V}} T_{\mathcal{G}}(w'|w) \cdot P(w'|\theta_c) \quad (15)$$

where $T_{\mathcal{G}}$ is the translation model generated from the target corpus $\mathcal{G}$.

Finally, to obtain the tailored profiles, we filter out all the concepts whose profile produces a distance above a given threshold. We show an excerpt of the tailored biochemistry residue profile, where the word protein dominates.

```
W011258494:  protein:0.883 method:0.030 sequence:0.0253
...
```

5.4. Measuring the semantic overlap of the KRs

Modeling the concepts in a KR as concept profiles gives us several advantages. For example, for each pair of KRs, we can estimate the level of redundancy between them by checking their associated concept profiles. Thus, to obtain the alignments, for each pair of concepts (c, c') such that $c \in KR$ and $c' \in KR'$ and $lex(c) \approx lex(c')$ we can estimate their semantic overlap with $D(\theta_c, \theta'_c)$ and a predefined threshold over it. Those pairs of concepts with similar lexica having a high overlap in their profiles are candidates to represent the same meaning.

5.5. Context-based semantic annotation

In this paper, we adopt the IR-based approach described in [7], which maps text chunks t to the KR lexicon strings of each concept c according to the following information-theoretical measure:

$$sim(t, c) = max_{s \in lex(c)} \frac{info(s \cap t) - info(t - s)}{info(s)}$$

The function $info(s) = \sum_{w \in s} -log(p(w|\mathcal{B}))$ estimates the information of a string s in terms of its probability in a background corpus (e.g., Wikipedia).

Notice that highly frequent words in the KR contribute little to the final score of the strings containing them. As a result, $sim(t, c)$ returns a list of candidate concepts for t with a normalized score between 0 and 1. This approach is similar to dictionary look-up approaches but it is flexible enough as it allows to select candidate concepts c whose $lex(c)$ better discriminates it and partially matches t.

To deal with the problem of ambiguous and wrong annotations we resort to the context-based validation of the candidate concepts. For this, we measure the distance between the local context of the annotation θ_{ann} and the tailored profile for the candidate concept $\theta_{c,\mathcal{G}}$, resulting in the final score $D(\theta_{ann}, \theta_{c,\mathcal{G}})$. To validate the annotation, we filter out all the concepts producing a distance above some threshold. In case of an ambiguous annotation, the selected concepts are:

$$argmin_{c' \in ann}\ D(\theta_{ann}, \theta_{c,\mathcal{G}})$$

The local context θ_{ann} for the annotation is obtained by taking a window of fixed size around the annotation and building its corresponding language model estimated via MLE.

5.6. Semantic search

The semantic search proposed in this paper relies on the distributions space where concepts, documents and query language models are placed. Basically, a semantic search consists of picking up a set of concepts from the KRs, building the corresponding query model, and selecting the nearest document models. Next subsections describe in detail this process.

5.6.1. Semantic representation of documents

Once the documents have been semantically annotated, they can be represented with the corresponding distribution of concepts involved in the annotations. In this way, each $d \in \mathcal{G}$ has associated a semantic model $\widehat{\theta}_d$, which is estimated as follows:

$$p(c|\widehat{\theta}_d) = \frac{tf(c, d)}{\sum_{c_i \in d} tf(c_i, d)} \tag{16}$$

This model clearly benefits the most frequent concepts, which are usually those with broader meanings. In order to capture the topicality of the concepts, we apply the parsimonious method previously described (Section 3.3), taking as background model the distribution of concepts in the target corpus $\mathcal{G}$. The resulting model θ^s_d is then used for indexing the document d.

5.6.2. Semantic query models

A semantic search (query) consists of the set of concepts $q = \{c_1, ..., c_k\}$ that best fit the user's information need. Without any prior knowledge about the relevance of these concepts w.r.t. the user requirements, we assume that the basic query model follows the uniform distribution, that is $p(c|\widehat{\theta}_q) = 1/|q|$. However, as the target corpus is biased towards very frequent concepts, we need to capture somehow the topicality of the query's concepts. Again, we apply the parsimonious smoothing to the query model to favor more specific concepts. In this case, we also use the concept distribution of the target corpus as

background model. The resulting model is denoted as θ_q^s.

As mentioned in the introduction, semantic search can take advantage from the KRs by expanding queries with their concept taxonomic relationships ($\preceq$). For any query, we can consider the *downwards expansion* of a query q as:

$$q^{\downarrow} = q \cup \{c_i \in \mathcal{C}_{KR} / c_i \preceq c\}$$

We can also consider the upwards expansion of a query q as:

$$q^{\uparrow} = q \cup \{c_i \in \mathcal{C}_{KR} / c \preceq c_i\}$$

and a combination of both expansions, represented with $q^{\updownarrow}$.

Now the problem is how to smooth the original query model in order to take into account the new expanded concepts. For this purpose, we use a smoothing operator based on random walks [42] following the regularization framework presented in [47]. Firstly, we define the affinity matrix M to embed the taxonomic relations involved in the query as follows:

$$M_{i,j} = \begin{cases} 1 & \text{if } i = j \text{ and } c_i \text{ has no parents} \\ \frac{1}{|parents(c_i)|} & \text{if } c_j \in parents(c_i) \\ \frac{1}{|children(c_i)|} & \text{if } c_j \in children(c_i) \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

From this matrix, we obtain the translation model $T_{\preceq}$ as follows:

$$T_{\preceq} = (1 - \delta) \cdot (I - \delta \cdot M)^{-1} \quad (18)$$

where δ is the diffusion factor (i.e., how much mass from the original query is diffused to the expanded concepts), and I is the identity matrix.

In this way, the model for the expanded query q' is generated by applying this translation model as follows:

$$p(c|\tilde{\theta}_{q'}) = \sum_{c' \in q'} T_{\preceq}(c|c') P(c'|\widehat{\theta}_q) \quad (19)$$

Finally, the semantic search is just performed by computing the distance $D(\tilde{\theta}_{q'}, \theta_d^s)$ over all the indexed documents d, ranking them from lower to higher values. The implementation details of this method are given in Section 6.7.

6. Experiments

We have performed several experiments in order to evaluate each of the phases of the proposed method. First, we describe the general setup in which the experiments take place. Then, for each experiment, we describe its objective and the specific datasets and resources used.

6.1. Datasets and characteristics of the KRs

For the experiments, we have considered a dataset, two large KRs, a pool of queries and four gold standards (GS) that involve several domains. The dataset used for annotation and semantic search, $WebRes$, is composed by metadata from 10,692 web resources for Life Sciences. As for the KRs, we have selected two well-known knowledge resources: UMLS®[10] and Wikinet[38]. We evaluate our semantic annotation method and compare it against others using the GS MSH-WSD[25], which is used by state-of-the-art disambiguation approaches and has been specifically designed to evaluate hard disambiguation cases over UMLS®. We have built the GSs, GS_{UMLS} and $GS_{Wikinet}$, to evaluate the performance of the semantic annotation over $WebRes$. For the semantic search evaluation, we have created a query pool that consists of descriptions of bioinformatics tasks and a GS, GS_{query}, to evaluate the retrieval results. All the datasets used are freely available[2].

Regarding the KRs, Table 1 shows the number of concepts of each KR, the size of their lexicon, the number of concept definitions and the number of "is-a" relationships. The characteristics of the annotation dataset and the three GSs will be explained in more detail in the experiments that make use of them.

In the previous resources, we distinguish two main domains that overlap: Biomedicine, which combines

[2] `http://krono.act.uji.es/TSASS`

KR	$\|C\|$	lex	def	$\preceq$
UMLS® 2012AB [10]	2,356,241	3,142,828	185,287	4,280,030
Wikinet* [38]	4,072,845	6,888,664	4,120,340	14,280,261

Table 1: Features of the KRs. *Only English lexicon.

vocabularies from Biology and Medicine, and Bioinformatics, which combines vocabularies from Biology and Computer Science. UMLS® and MSH-WSD are both located in the Biomedicine domain, whereas Wikinet does not have a specific location because it covers several domains but with low specificity. The $WebRes$ dataset overlaps only partially with the Biomedicine and Bioinformatics domain, and GS_{UMLS}, $GS_{Wikinet}$ and GS_{query} are located inside the $WebRes$ and overlapping with the two main domains. This heterogeneous setup makes semantic annotation w.r.t. the KRs especially hard because the $WebRes$ dataset overlaps only partially with the reference KRs. The aim of the following experiments is to show that the context-validated semantic annotations using profiles based on statistical language models and the tailoring (both of the KR and the profiles) in such an heterogeneous scenario improves semantic annotation and therefore, semantic search.

6.2. Tailoring of the KRs

The tailoring of a KR (Section 5.1) consists in selecting those concepts from the KR that are semantically related to the target corpus. This filtering process can reduce the overhead of the semantic annotation process, specially when the KR is very large. We have applied the tailoring to both UMLS® and Wikinet. As a result, we obtain 171,274 concepts for UMLS® and 510,390 for Wikinet. Recall that this process selects only concepts whose definition is more likely to be generated from the corpus than the KR. Therefore, the tailoring depends on the number of definitions of the KR. The proportion of the concepts selected w.r.t. to the definitions for UMLS® is 92%, which indicates that UMLS® is well-suited to the corpus and the tailoring process discards few concepts. However, for Wikinet this proportion is only 12.3%, which means it contains a lot of noise (i.e., concepts not related to the target corpus) that has been removed through the tailoring process. Therefore, from now on we use the tailored version of Wikinet, Wikinet^T, and UMLS® without tailoring.

6.3. Concept profile evaluation

The KR concept profiles play a crucial role in the semantic annotation process, as they serve to disambiguate ambiguous semantic annotations (see Definition 3.3). In this section we evaluate the quality of the concept profiles by means of two experiments.

In the first experiment, we compare our approach for context-validated semantic annotation with state-of-the art WSD methods. Recall that knowledge-based WSD methods deal with the problem of selecting one of the senses of a word from an inventory of words and their senses, whereas our method is thought to perform an unsupervised, full-fledged semantic annotation. The phase that resembles WSD is the context validation phase, where we have a profile based on translation models for each candidate concept and compare it with the context around the annotation to select valid concepts for such annotation.

We use the MSH-WSD dataset [25] for evaluating this phase. This corpus contains 203 strings that are associated with more than one possible MeSH code in the UMLS® Metathesaurus (106 of these are ambiguous abbreviations, 88 ambiguous terms and 9 a combination of both). The corpus contains up to 100 examples for each possible sense, and a total of 37,888 examples of ambiguous strings taken from Medline.

We evaluate the context validation phase with both the concept profiles (TrM) and the extended concept profiles (TrMExt) described in Section 5.2. The parameters α, β and γ have been empirically set to 0.45, 0.50, 0.40, respectively. Performance is compared against various alternative approaches. Accuracy results of the experiments are shown in Table 2. Both MRD[33] and 2-MRD[34] are unsupervised

approaches based on building concept vector profiles normalized by IDF and comparing them with the context vector using cosine similarity. PPR [3] is also unsupervised and relies on a graph-based algorithm similar to the page rank that converts UMLS® into a graph where the possible meanings of ambiguous words are nodes and relations between them are edges. AEC [23] and UB [11] are supervised learning algorithms that alleviate the problem of requiring manually annotated training data by querying Medline documents. Our methods present very good scores against unsupervised approaches of the literature and near to semi-supervised ones. Moreover, the extended version improves results over the original one, that is, including information about the concept hierarchy in the profiles helps disambiguation.

The aim of the second experiment is to evaluate our method for context-validated semantic annotation in the web resource discovery domain, which is hampered by the heterogeneity of data and where the use of general words introduces a lot of ambiguity. Thus, we have built up a dataset of 2,260 web resources from BioCatalogue [8], which is a popular registry in the Life Sciences domain. The web resources metadata registered in this repository consists of well-defined fields, such as categories and tags, and textual descriptions.

To evaluate the semantic annotation over the previous dataset, we have manually created two GSs for the two KRs, GS_{UMLS} and $GS_{Wikinet}$, with those annotations matching a single word, as single word concepts are much prone to ambiguity and errors. GS_{UMLS} contains 11,041 single-word semantic annotations and $GS_{Wikinet}$ contains 5,386.

We have evaluated five configurations of our semantic annotation method: context-free, context-validated using the TrM method for the concept profiles, context-validated using the TrMExt method, and the tailored versions of the last two methods, that is, where the concept profiles have been filtered as indicated in Section 5.3. The threshold used to filter concept profiles is 0. First, we present Table 3, which shows the average number of concept profiles in the original and tailored versions. As observed, the reduction of the number of concept profiles in the tailored versions is very significant, all of them reaching a reduction around 90% or more.

Table 4 shows the results of the semantic annotation evaluation for the previous five configurations using both UMLS® and Wikinet^T. We use the standard measures precision, recall, F measure and accuracy to evaluate the resulting annotations.

The results show that all the proposed methods improve the results of the context-free annotation method. In general, we observe that the extended versions of the methods do not improve results in any of the cases, which means that the information provided by the concept hierarchy is not decisive for validation in this dataset. This may be due to the mismatch of domains between the annotation dataset $WebRes$ w.r.t. both UMLS® and Wikinet. In this case, including information about the hierarchy in the concept profiles seems to introduce noise that does not help disambiguation, as opposed to the performance of the extended version in MSH-WSD (see Table 2), where the domains of the GS and the annotation dataset are the same.

The tailored versions suffer a decrease in all the measures but results are still comparable to state-of-the-art WSD approaches. The lower performance is more noticeable in UMLS®, specially w.r.t. the recall. This indicates that the concept tailoring in UMLS® may be too aggressive, and potentially good concepts are being filtered, whereas the concept tailoring in Wikinet^T seems to work better, as it is able to keep performance while reducing the number of concept profiles. Notice that in this dataset, resource descriptions focus on software aspects and, therefore, the contexts are not related to biological terms. Still, the reduction in the number of concept profiles of the tailored versions (see Table 3) make them an ideal choice when dealing with huge amounts of concept profiles.

From now on, when we refer to the semantic annotation process or the concept profiles, we mean the context-validated semantic annotation using the concept profiles generated by the method TrM^T, which is the method that offers the best trade-off between all the measures.

MRD	2-MRD	PPR	AEC	UB	TrM	TrMExt
0.8070	0.7799	0.7860	0.8383	0.8319	0.8010	0.8212

Table 2: WSD evaluation results in terms of accuracy on MSH-WSD dataset. MRD stands for Machine Readable dictionary, 2-MRD stands for 2nd Order Co-occurrence MRD, PPR stands for Personalised Page Rank, AEC stands for Automatic Extracted Corpus, UB stands for Uniform Bias, TrM stands for Translation Model and TrMExt stands for Translation Model Extended.

KRs	TrM	$\mathbf{TrM}^T$	TrMExt	$\mathbf{TrMExt}^T$
UMLS®	1416.5	95 (93.3%)	3109.7	164 (94.7%)
Wikinet^T	2303	244 (89.4%)	3141.8	276 (91.2%)

Table 3: Average size of the concept profiles used for the validation of semantic annotations in each method.

KRs	Meas.	Ctxt-free	TrM	TrMExt	$\mathbf{TrM}^T$	$\mathbf{TrMExt}^T$
UMLS®	P	0.721	0.893	0.875	0.850	0.842
	R	0.939	0.799	0.799	0.701	0.711
	F	0.778	0.815	0.805	0.725	0.727
	Acc	0.744	**0.838**	0.813	0.768	0.758
Wikinet^T	P	0.637	0.843	0.840	0.822	0.819
	R	0.994	0.899	0.892	0.880	0.880
	F	0.714	0.847	0.843	0.823	0.821
	Acc	0.639	**0.869**	0.867	0.847	0.847

Table 4: Macro average Precision (P), recall (R), F-measure (F) and accuracy (Acc) of semantic annotations with different configurations of context validation.

6.4. Alignments between KRs

In this experiment, we measure the overlap between Wikinet^T and UMLS® by obtaining a set of concept alignments. For each pair of concepts (c, c') such that $c \in C_{UMLS}$ and $c' \in C_{Wikinet^T}$ and $lex(c) \approx lex(c')$, we estimate their semantic overlap by comparing their profiles $D(\theta_c, \theta_{c'})$ and filtering out those below a predefined threshold. As a result, we obtain a set of 6,058 alignments. Notice that the resulting set of alignments is rather small, which indicates both KRs are complementary. From this set, we distinguish the alignments between concepts of only one word, S_{one}, (91 alignments), and concepts with more than one word, S_n, (5,967 alignments). The set S_{one} was manually assessed and has a precision of 54%, whereas for the set S_n we manually assessed a hundred random samples, resulting in a precision of 87%. This confirms the hypothesis that short-lengthed concepts are more difficult to disambiguate and, in this case, to correctly align.

6.5. Semantic annotation evaluation

In this experiment we evaluate the impact of the context-free vs. context-validated annotations. The dataset that will be annotated is composed by metadata from 10,692 web resources, of which 6,226 are related to the Life Sciences domain and 4,466 are of general domains registered in Wikipedia. We have downloaded the metadata of the Life Sciences web resources from BioCatalogue (more than 2,200 web resources), myExperiment [20] (more than 2,000 workflows), and SSWAP [19] (more than 2,700 web resources). With respect to the web resources registered in Wikipedia, we have considered those entries that describe web resources, independently of their domain. In order to select those entries, we have applied category filters and lexical patterns to identify expressions related to web resources, e.g., "is a web service", "is a database", etc.

Table 5 shows the number of different concepts in the annotations of the dataset, the total number of annotations, and their ambiguity[3] in context-free versus context-validated annotations. The experiments are reproduced for two configurations of the KRs, with and without tailoring of Wikinet. We observe that the number of context-validated annotations has been reduced to roughly a third w.r.t the number of context-free annotations. However, the most remarkable fact is that the ambiguity of annotations is much higher in context-free annotations, and this affects the semantic search as will be demonstrated in the next section. In the context-validated annotations, with the TrM^T method we reduce the ambiguity and also fewer annotations are produced. Similarly, regarding the semantic annotation using the tailored version of Wikinet, we observe that the ambiguity is reduced and also fewer annotations are produced, which may affect recall. However, as shown in the previous Table 4, the trade-off between precision and recall when using tailoring over the KR (i.e., Wikinet^T) and over the method for profile generation (i.e., TrM^T) is good.

[3]Ambiguity is calculated as the percentage of annotations that have been assigned more than one sense.

6.6. Time performance evaluation

The proposed context-validated semantic annotation process does not imply a computational overhead as many WSD methods do. Table 6 shows the time performance of each of the components for the semantic annotation of the 10,692 web resources dataset. The first three phases are done only once off-line. In the profile generation phase, we distinguish between the normal and the extended version because in the extended version all the direct parents and children of the concept are considered for generating the profile, thus incurring in extra time. We also distinguish between UMLS® and Wikinet because the performance is significantly different. While the profile generation is faster in Wikinet, probably because of shorter concept labels and definitions, it happens the opposite for the extended version. This is due to the fact that the average number of direct parents and children in Wikinet is three times more than for UMLS®, making the extended version in Wikinet slower. In the profile tailoring phase we also make the distinction because extended profiles are considerably larger, thus affecting the tailoring performance. Finally, it is worth mentioning that both the context-free and the context-validated annotations have a

	Context-free			Context-validated		
				TrM/TrM^T		
KRs	**Conc.**	**Ann.**	**Amb.**	**Conc.**	**Ann.**	**Amb.**
Wikinet^T UMLS®	24,612	374,623	41.67%	14,678/14,334	111,085/105,094	12.62%/11.72%
Wikinet UMLS®	39,623	399,687	41.59%	25,777/25,448	130,541/124,484	14.24%/13.51%

Table 5: Results of the semantic annotation process using different configurations of the KRs for context-free vs. context validated annotations. ^{T} means tailored version.

similar performance, which we measure in annotated documents per second.

6.7. Semantic search evaluation

The experiments carried out to perform the evaluation of the semantic search consist in the execution of a set of heterogeneous queries (i.e., task description examples) over the dataset of 10,692 web resources. These queries capture different ways to describe bioinformatics tasks (see Table 7), thus reflecting the variability in the users' information needs. The query pool was created by selecting more than 250 short descriptions extracted from other Life Sciences resource catalogues such as OBRC[4] (Online Bioinformatics Resource Collection) and ExPaSy[5] (SIB Bioinformatics Resource Portal). Thus, we have selected as queries the short descriptions of the resources registered on these catalogues. All the queries have been semantically annotated and expanded with related concepts in the KR, as described in Section 5.6.2. As a result, each query has associated a semantic query model. To evaluate the retrieval results, we have built an assessment dataset, GS_{query}, with relevant descriptions associated to each task. This dataset was set-up by selecting predefined categories and tags from the target catalogues which are relevant to each task.

In these experiments, we implemented a search engine based on language models, indexed under a traditional inverted file [32]. Thus, indexed descriptions are retrieved and ranked according to their similarity to the query, in this case calculated with the distance between models (Section 5.6.2). On top of this basic search engine, we implemented both a keyword-based and a semantic-based search method. The former defines language models directly from words, whereas the latter uses the semantic models defined in Section 3.3. The keyword-based method is used as baseline to demonstrate that semantic annotations improve the retrieval effectiveness. Table 8 shows the precision at 5, 10 and 20, and the Mean Average Precision (MAP) for the query results using the keyword-based method evaluated against GS_{query}.

Topic	**P@5**	**P@10**	**P@20**	**MAP**
T_1	0.63	0.59	0.59	0.21
T_2	0.59	0.61	0.62	0.33
T_3	0.8	0.74	0.68	0.18
T_4	0.79	0.81	0.75	0.45
T_5	0.77	0.78	0.81	0.21
T_6	0.83	0.78	0.79	0.22
T_7	0.53	0.45	0.37	0.22
T_8	0.6	0.6	0.58	0.13
Average	0.69	0.67	0.65	0.24

Table 8: Precision at n (P@n) for the top-5, top-10, and top-20 results, and MAP measure for the keyword-based search.

We have evaluated the semantic-based search using different configurations in order to evaluate the impact of using tailored KRs and contexts on the retrieval results. Table 9 shows the precision at 5, 10 and 20, and the MAP measure of the query results using the different configurations. As it can be noticed, in general the semantic search presents higher precision scores at the first top-ranked positions than the keyword-based search using smaller models (39,253 terms against 14,678 concepts in the best performance configuration, tailoring with TrM concepts profiles). Next, we analyze in detail the different configurations evaluated in these experiments.

[4]http://www.hsls.pitt.edu/obrc/
[5]http://expasy.org

Phase		UMLS®	Wikinet
KR tailoring		-	0,001 c/sec
Profile generation	TrM	1,7 c/sec	5,7 c/sec
	TrMExt	0,83 c/sec	0,38 c/sec
Profile tailoring	TrM	0.052 c/sec	
	TrMExt	0.082 c/sec	
Context-free annotation		0.48 d/sec	
Context-validated annotation		0.51 d/sec	

Table 6: Performance in concepts per second (c/sec) and documents per second (d/sec) of each of the phases of the semantic annotation for the web resources dataset (1 CPU).

Task	Description	N. of queries
T_1	Search proteins with a functional domain	14
T_2	Search similar sequences	16
T_3	Analyze phylogeny	14
T_4	Align sequences	24
T_5	Analyze transgenic model organism	31
T_6	Predict structure	30
T_7	Protein identification and characterization	12
T_8	Find genes with functional relationships	42

Table 7: Bioinformatics base tasks considered for evaluation.

With respect to the consideration of the context during the semantic annotation, we have executed the queries without validation, and validating annotations with the two best configurations of concepts profiles, the TrM concept profiles model and its tailored version TrM^T. The results show that the precision scores are better when validating the annotations contexts. In contrast, the MAP measure is better when not considering the context because the recall is higher when all senses are included. Regarding the results for the two different context models, there is not much difference between them, although the tailored version obtains slightly worse precision at the first positions.

Regarding the tailoring of KRs, the use of a tailored KR reduces considerably the semantic index and also the ambiguity of the annotations (see Table 5), while the results are not affected by the reduction of annotations. Moreover, the precision at the top-ranked positions is slightly higher when using the tailored version of Wikinet.

Finally, we have also analyzed the impact of the query expansion on the retrieval results. We have executed the queries for the best configuration in Table 9 but without expanding the query. The resulting precision scores are slightly lower (P@5=0.74, P@10=0.7, P@20=0.66, MAP=0.19).

In conclusion, semantic search obtains better results than the keyword-based search using considerably much smaller indexes. We have demonstrated that using tailored KRs in the semantic search reduces the size of indexes without losing accuracy in the retrieval results. Moreover, the consideration of contexts in the semantic annotation process reduces the ambiguity of the resulting annotations, providing in this way more accurate results. Therefore, we have demonstrated that the tailored context-based semantic annotation obtains good results in the semantic search with higher efficiency and lower overhead.

7. Related Work

7.1. Semantic search

Semantic search aims to identify the user's intent and to retrieve the documents that best fit this intent, independently of the terms provided by the

		Context-free				Context-validated			
						TrM/TrM^T			
KRs	**Task**	**P@5**	**P@10**	**P@20**	**MAP**	**P@5**	**P@10**	**P@20**	**MAP**
Wikinet^T UMLS®	T_1	0.73	0.73	0.65	0.16	0.69/0.66	0.6/0.62	0.53/0.55	0.15/0.15
	T_2	0.77	0.74	0.71	0.29	0.78/0.74	0.68/0.69	0.69/0.69	0.25/0.24
	T_3	0.73	0.71	0.66	0.43	0.86/0.84	0.84/0.8	0.75/0.68	0.36/0.33
	T_4	0.85	0.87	0.85	0.35	0.76/0.82	0.8/0.71	0.78/0.66	0.36/0.33
	T_5	0.87	0.85	0.81	0.22	0.83/0.85	0.87/0.85	0.85/0.84	0.2/0.19
	T_6	0.85	0.82	0.77	0.15	0.87/0.89	0.82/0.83	0.78/0.8	0.1/0.1
	T_7	0.35	0.27	0.23	0.09	0.55/0.48	0.46/0.44	0.35/0.35	0.16/0.15
	T_8	0.6	0.63	0.6	0.13	0.74/0.73	0.71/0.71	0.66/0.66	0.11/0.12
	Avg.	0.72	0.7	0.66	0.23	0.76/0.75	0.73/0.73	0.68/0.68	0.2/0.19
Wikinet UMLS®	T_1	0.63	0.6	0.57	0.15	0.6/0.6	0.55/0.55	0.53/0.55	0.14/0.14
	T_2	0.79	0.7	0.68	0.23	0.67/0.67	0.59/0.58	0.58/0.59	0.22/0.22
	T_3	0.71	0.61	0.59	0.42	0.86/0.67	0.82/0.71	0.69/0.67	0.25/0.22
	T_4	0.93	0.9	0.88	0.35	0.82/0.86	0.82/0.86	0.79/0.8	0.24/0.24
	T_5	0.83	0.85	0.82	0.22	0.81/0.83	0.82/0.84	0.83/0.85	0.2/0.2
	T_6	0.89	0.84	0.83	0.18	0.83/0.85	0.8/0.81	0.82/0.83	0.12/0.12
	T_7	0.35	0.27	0.23	0.08	0.53/0.48	0.36/0.37	0.28/0.29	0.14/0.14
	T_8	0.64	0.64	0.63	0.14	0.8/0.79	0.78/0.75	0.74/0.72	0.11/0.12
	Avg.	0.72	0.68	0.65	0.22	0.74/0.72	0.69/0.68	0.66/0.66	0.18/0.17

Table 9: Precision at n (P@n) for the top-5, top-10, and top-20 results, and Mean Average Precision (MAP) of search results with different annotation configurations.

user in the specification of her information needs. The basis of the semantic search are the conceptual representation of the information, however most approaches only use this representation to expand the user's query. There are approaches that assume that the documents are semantically indexed, either manually or automatically, with concepts from a known KR. For example, PubMed expands the query using MeSH terms and performs a boolean search based on these terms. [36] uses pseudo-relevance feedback to include the MeSH terms of the top-k relevant documents, retrieved by an initial search, to the user's query, and then performs a keywords-based retrieval. Other approaches do not consider the conceptual representation of documents, and only use the knowledge in KRs to expand the query. For example, [24] uses the concepts representing the user's intent to expand the query with the terms associated to those concepts in the KR, then the retrieval is based on keyword matching. Currently, few approaches consider the conceptual representation of both the user's requirements and the documents. There are approaches in which the documents are semantically represented as entity-relationship graphs and make use of graph-based query languages to perform semantic search [26, 15]. In the Life Sciences domain, SADI [46] performs semantic search via SPARQL queries of web services that have been previously semantically represented in RDF. These approaches require the documents to be in RDF format, which is not very frequent in general, even through there is current research towards this direction [17].

7.2. Semantic annotation

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 KRs ranging from lexicons, terminologies and thesauri to expressive ontologies, are publicly accessible and ready to be used for annotation purposes such as *dbpedia*[6], *yago*[7], *freebase*[8] and *schema.org*[9]. Specially in the biomedical domain we can find several lexical/ontological specialized resources such as MeSH, SNOMED, UMLS® and BioPortal among others. The use of knowledge-based semantic annotation can have a great impact on semantic search, as both the user query and the documents are represented in a conceptual space.

[6] `http://es.dbpedia.org/`
[7] `http://www.mpi-inf.mpg.de/yago-naga/yago/`
[8] `http://www.freebase.com/`
[9] `http://schema.org`

For semantic annotation, 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*[37], *The Wiki Machine*[10], *AlchemyAPI*[11] and *Open Calais*[12], for annotating general-purpose entities, or *MetaMap*[5] and *Whatizit*[44] for annotating biomedical entities.

Most of the unsupervised semantic annotation methods rely on a dictionary look-up strategy. Basically, it consists of finding occurrences of concept strings in a text chunk by applying strict string matching. To allow some small variations in the matching (e.g., plural forms), concept strings can be translated into regular expressions, which are applied to the text chunks to obtain the mappings [44, 13]

Other approaches adopt an information retrieval (IR) strategy [5, 7]. Basically, it consists of viewing the text chunk T as a query, and the concept strings as documents to be retrieved. This strategy notably increases the recall since it disregards the order and continuity of the matched words. To allow more flexibility in the matching, the query generated by T can be expanded with the variants of each word w_i (e.g., plurals, hyphenation, abbreviations, etc.) to perform the retrieval.

The majority of these approaches still perform poorly with ambiguous annotations. Some of them make use of contextual information (e.g., words around the annotation) to improve disambiguation. Still, results are not satisfactory mainly because of two reasons: 1) the KR does not have the appropriate sense and 2) the method for comparing the contexts is too trivial. This issue has been thoroughly studied by WSD methods, which are explained in the following section.

7.3. Word sense disambiguation

WSD is one of the key tasks in natural language processing (NLP) applications. Although WSD is focused on choosing the right sense for each word in a sentence, it can be somehow extrapolated to the problem of disambiguating semantic annotations. More specifically, knowledge-based WSD methods [39] can be adapted to choose the concepts that best fit to the text where they are identified. Most knowledge-based WSD methods are almost unsupervised as they mainly rely on the information provided by the lexical knowledge resource (mainly WordNet and its variants). Some additional heuristics such as the most frequent sense (MFS) are often included to help in the final decisions, hence the almost. Former approaches to knowledge-based WSD consisted of variations of the Lesk algorithm [29], which basically compares the glosses of the senses provided by the KR with the words surrounding the word to be disambiguated. In this way, the disambiguation problem consists of measuring the overlap between the term-vectors associated to each concept (also called topic signatures [2]) in the KR and the term-vector of the target word context, and then to select the concept that gives the highest score. These approaches assume that the richer the topic signatures are the better is the chance to choose the right one. Thus, in [2] term-vectors are built by querying Google with monosemous synonyms or hyponyms of each concept, and then weighting them with a tf-idf scheme. In [22] a similar approach is proposed to build term-vectors for UMLS® concepts by querying PubMed with MeSH terms. In [4] term-vectors are built with the words of the glosses of the hyperonyms and hyponyms of each word sense, also weighted with a tf-idf scheme. More recent approaches attempt to extract the knowledge encapsulated in the KR to get more evidence for decision making. For example, in [4] implicit relations are found by comparing the topic signatures of all the senses involved in a sentence. In [12] topic signatures are used to discover relations between word senses. Such discovered relations have shown useful for WSD when applying random walks techniques over the resulting word sense graphs [40, 16]. In the context of semantic annotations with arbitrary KRs, knowledge-based methods are difficult to apply mainly because they have been developed taking advantage from the particular characteristics of the lexical KR they are aimed at, such as the rich WordNet relations, or the link structure of Wikipedia

[10] `http://thewikimachine.fbk.eu`
[11] `http://www.alchemyapi.com/`
[12] `http://www.opencalais.com/`

[35]. Moreover, they are not aimed at validating the generated annotations but at choosing one of the existing senses, which can lead to wrong annotations if the right sense is not covered by the KR. Our approach for validation is inspired in the Lesk principle combined with topic signatures. However, we rely on a statistical framework to generate the concept language models and to compare them with the corpus contexts, also represented as language models.

8. Conclusions

In this paper we have proposed a novel method for semantic annotation and search based on statistical language models. Our main hypothesis is that reconciling the vocabulary in the KRs and the target corpus can lead to more precise and useful annotations. We achieved such reconciliation by means of statistical translation models, which enable to define rich language models for both the KR concepts and the corpus contexts. From the experiments we can draw several conclusions:

- Coarse tailoring is useful for very large and heterogeneous KRs like Wikinet, since we can easily reject those parts of the KR that have nothing to do with the target corpus. However, more specialist KRs like UMLS® are much more homogeneous and take little advantage from the coarse tailoring.

- In some scenarios it is necessary to combine more than one KR in order to get a proper coverage of the target corpus. Otherwise, semantic search will be less effective than keyword-based search. In our experiments, web resource catalogues combine computer science and biomedical terminologies, which cannot be properly covered with a unique KR. We have shown that UMLS® and Wikinet complement each other quite well for this domain.

- Language models generated with translation models have proved very useful in tailoring and validating semantic annotations.

- Results show a dramatic reduction in the number of obtained annotations (and therefore the size of the semantic search structures) at the same time that precision increases with little lost in recall.

As future work, there are several interesting research lines derived from this work. First, we plan to study new approaches for concept profile construction that combine topic-based models like Latent Diritchlet Allocation (LDA) [9] with the translation models proposed in this paper. LDA has been shown very useful in WSD tasks [30] and provides a statistical framework that captures word co-occurrence patterns at collection level. We also plan to apply topic-based models for performing semantic searches. This idea has been previously explored in [43] by using context-free annotations with good results. The main limitation of this approach is that topics must be defined a priori and they are dependent on the application domain. We will investigate how to automatically generate topics of interest from the profiles of the KRs and the corpus at hand. Finally, we will study how to take more profit from the KR taxonomic relationships in order to enhance the KRs translation models and the generated concept profiles. Moreover, we will consider the construction of the graph of concepts induced by their contexts relationships similarly to some knowledge-based graph WSD approaches [40].

References

[1] PubMed. `http://www.ncbi.nlm.nih.gov/pubmed`.

[2] E. Agirre and O. L. de Lacalle. Publicly Available Topic Signatures for all WordNet Nominal Senses. In *LREC*. European Language Resources Association, 2004.

[3] E. Agirre, A. Soroa, and M. Stevenson. Graph-based word sense disambiguation of biomedical documents. *Bioinformatics*, 26(22):2889–2896, 2010.

[4] H. Anaya-Sanchez, A. Pons-Porrata, and R. Berlanga. TKB-UO: using sense clustering

for WSD. In *SemEval 07: Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 322–325, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[5] 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/`.

[6] R. Berlanga, E. Jiménez-Ruiz, and V. Nebot. Exploring and linking biomedical resources through multidimensional semantic spaces. *BMC Bioinformatics*, 13(S-1):S6, 2012.

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

[8] J. Bhagat, F. Tanoh, E. Nzuobontane, T. Laurent, J. Orlowski, M. Roos, K. Wolstencroft, S. Aleksejevs, R. Stevens, S. Pettifer, R. Lopez, and C. A. Globe. BioCatalogue: a universal catalogue of web services for the life sciences. *NAR*, 2010.

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, Jan. 2003.

[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] W. Cheng, J. Preiss, and M. Stevenson. Scaling up WSD with Automatically Generated Examples. In *Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*, BioNPL '12, pages 231–239, 2012.

[12] M. Cuadros and G. Rigau. KnowNet: building a large net of knowledge from the web. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 161–168, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[13] 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.

[14] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[15] S. Elbassuoni and R. Blanco. Keyword search over rdf graphs. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 237–242, New York, NY, USA, 2011. ACM.

[16] A. S. Eneko Agirre, Oier Lopez de Lacalle. Random Walks for Knowledge-Based Word Sense Disambiguation. *Computational Linguistics*, In Press:1–48, 2013.

[17] L. Garcia Castro, C. McLaughlin, and A. Garcia. Biotea: RDFizing PubMed Central in Support for the Paper as an Interface to the Web of Data. *Biomedical semantics*, 4(Suppl 1):S5, 2013.

[18] L. García-Moya, H. Anaya-Sánchez, and R. Berlanga. Combining Probabilistic Language Models for Aspect-Based Sentiment Retrieval. In R. A. Baeza-Yates, A. P. de Vries, H. Zaragoza, B. B. Cambazoglu, V. Murdock, R. Lempel, and F. Silvestri, editors, *Proceedings of the 34th European Conference on IR Research, ECIR 2012*, volume 7224 of *Lecture Notes in Computer Science*, pages 561–564. Springer, 2012.

[19] D. D. Gessler, G. S. Schiltz, G. D. May, S. Avraham, C. D. Town, D. Grant, and R. T. Nelson. SSWAP: A Simple Semantic Web Architecture and Protocol for semantic web services. *BMC Bioinformatics*, 10:309, 2009.

[20] C. A. Goble, J. Bhagat, S. Aleksejevs, D. Cruickshank, D. Michaelides, D. Newman, M. Borkum, S. Bechhofer, M. Roos, P. Li, and D. De Roure. myExperiment: a repository and social network for the sharing of bioinformatics workflows. *Nucleic Acids Research*, 38(suppl 2):W677–W682, 2010.

[21] D. Hiemstra, S. Robertson, and H. Zaragoza. Parsimonious language models for information retrieval. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 178–185, New York, NY, USA, 2004. ACM.

[22] A. Jimeno-Yepes and A. R. Aronson. Knowledge-based biomedical word sense disambiguation: comparison of approaches. *BMC Bioinformatics*, 11:569, 2010.

[23] A. Jimeno-Yepes and A. R. Aronson. Knowledge-based biomedical word sense disambiguation: comparison of approaches. *BMC Bioinformatics*, 11:569, 2010.

[24] A. Jimeno-Yepes, R. Berlanga, and D. Rebholz-Schuhmann. Ontology refinement for improved information retrieval. *Information Processing & Management*, 46(4):426 – 435, 2010. Semantic Annotations in Information Retrieval.

[25] A. J. Jimeno-Yepes, B. T. McInnes, and A. R. Aronson. Exploiting mesh indexing in medline to generate a data set for word sense disambiguation. *BMC Bioinformatics*, 12:223, 2011.

[26] G. Kasneci, F. M. Suchanek, G. Ifrim, M. Ramanath, and G. Weikum. Naga: Searching and ranking knowledge. In G. Alonso, J. A. Blakeley, and A. L. P. Chen, editors, *ICDE*, pages 953–962. IEEE, 2008.

[27] J. Lafferty and G. Lebanon. Diffusion Kernels on Statistical Manifolds. *J. Mach. Learn. Res.*, 6:129–163, Dec. 2005.

[28] 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.

[29] M. Lesk. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*, SIGDOC '86, pages 24–26, New York, NY, USA, 1986. ACM.

[30] L. Li, B. Roth, and C. Sporleder. Topic models for word sense disambiguation and token-based idiom detection. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (ACL '10)*, pages 1138–1147, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[31] J. Lin and C. Dyer. *Data-Intensive Text Processing with MapReduce*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2010.

[32] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[33] B. T. McInnes. An unsupervised vector approach to biomedical term disambiguation: Integrating umls and medline. In E. Arisoy, K. Inoue, and W. Maier, editors, *ACL (Student Research Workshop)*, pages 49–54. The Association for Computer Linguistics, 2008.

[34] B. T. McInnes. *Supervised and Knowledge-based Methods for Disambiguating Terms in Biomedical Text using the UMLS and MetaMap*. PhD thesis, University of Minnesota, Minneapolis, MN, USA, 2009.

[35] O. Medelyan, D. Milne, C. Legg, and I. H. Witten. Mining meaning from Wikipedia. *Int. J. Hum.-Comput. Stud.*, 67(9):716–754, Sept. 2009.

[36] E. Meij, D. Trieschnigg, M. de Rijke, and W. Kraaij. Conceptual language models for domain-specific retrieval. *Inf. Process. Manage.*, 46(4):448–469, July 2010.

[37] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, I-Semantics '11, pages 1–8, New York, NY, USA, 2011. ACM.

[38] V. Nastase and M. Strube. Transforming Wikipedia into a large scale multilingual concept network. *Artif. Intell.*, 194:62–85, Jan. 2013.

[39] R. Navigli. Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2), 2009.

[40] R. Navigli and M. Lapata. An Experimental Study of Graph Connectivity for Unsupervised Word Sense Disambiguation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(4):678–692, 2010.

[41] V. Nebot and R. Berlanga. Exploiting Semantic Annotations for Open Information Extraction: an experience in the biomedical domain. *Knowl. Inf. Syst.*, 2013.

[42] J. R. Norris. *Markov Chains (Cambridge Series in Statistical and Probabilistic Mathematics)*. Cambridge University Press, July 1998.

[43] M. Pérez, R. Berlanga, I. Sanz, and M. J. Aramburu. A semantic approach for the requirement-driven discovery of web resources in the life sciences. *Knowl. Inf. Syst.*, 34(3):671–690, 2013.

[44] 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.

[45] M. Sun, G. Lebanon, and P. Kidwell. Estimating probabilities in recommendation systems. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61(3):471–492, 2012.

[46] M. D. Wilkinson, B. Vandervalk, and L. McCarthy. The Semantic Automated Discovery and Integration (SADI) Web service Design-Pattern, API and Reference Implementation. *Journal of Biomedical Semantics*, 2:8, 2011.

[47] D. Zhou and B. Schlkopf. A regularization framework for learning from graph data. In *ICML Workshop on Statistical Relational Learning*, pages 132–137, 2004.