MeDetect: A LOD-based System for Collective Entity Annotation in Biomedicine

Li Tian[1], Weinan Zhang[2], Antonis Bikakis[2], Haofen Wang[1], Yong Yu[1], Yuan Ni[3], Feng Cao[3]
[1]*Shanghai Jiao Tong University;* [2]*University College London;* [3]*IBM China Research Laboratory*
{tianli,whfcarter,yyu}@apex.sjtu.edu.cn; {ucabwz0,a.bikakis}@ucl.ac.uk; {niyuan,caofeng}@cn.ibm.com

***Abstract*—With the ever-growing use of textual biomedical data, domain entity annotation has become very important in biomedicine. Previous works on annotating domain entities from biomedical references suffer from several issues, such as a data flexibility problem, language dependency, and limitations with respect to word sense disambiguation. Meanwhile, the Linked Open Data (LOD) Initiative aims at interlinking data from various open knowledge bases. The numbers of entities and properties describing semantic relationships between entities within the linked data cloud have become very large. In this paper, we propose a knowledge-incentive approach for entity annotation in biomedicine, and present MeDetect, a prototype system that we developed based on this approach. With this approach, we overcome the problems of previous works using LOD-based collective annotation. Finally, we present the results of experiments that verify the effectiveness and efficiency of our approach.**

Keywords—Domain Entity Annotation, Linked Open Data, Bio-Informatics

I. INTRODUCTION

Entity annotation aims at discovering entities in references automatically. It is quite useful for many tasks including information extraction, classification, text summarization, question answering, and literature-based knowledge discovery. Recently, with the ever-growing supply of textual biomedical data, annotating domain entities has become important in biomedicine. A prominent example is MetaMap [4,5], which leverages symbolic, natural-language processing (NLP) to map biomedical text to the UMLS Metathesaurus or, equivalently, to discover Metathesaurus concepts referred to in text. It has been used as one of the foundations of NLM's Medical Text Indexer (MTI) [6] for semantic information retrieval on biomedical literature. Moreover, NLM uses MetaMap to build RIDeM (Repository for Informed Decision Making) with various components, such as InfoBot to link evidence with patient records, HDiscovery for annotation of clinical research publications, and CQA 1.0 for clinical question answering.

On the other hand, the Web is developing from a Web of documents to a Web of data. In the last few years, the amount of structured data available on the Web has been increasing rapidly. Currently, there are billions of triples publicly available in 295 Web data sources[1] of different domains written in a variety of languages[2]. These data sources are becoming more tightly interrelated as the number of links in the form of mappings is also growing. The process of interlinking open data sources is actively pursued within the Linked Open Data (LOD) [9] initiative, a grassroots community effort supported by W3C. A recent effort of embracing linked data principles in biomedicine is Linked Life Data [2], a semantic data integration platform including more than 1 billion entities. Another prominent example is Linked Open Drug Data [3, 14]. It contains drug related data sources, such as DrugBank, DailyMed, and Diseasome, and interlinks with other data sources from the domain of life science. Compared to UMLS, the numbers of entities and properties describing relationships between entities within the linked data cloud are much larger. Moreover, the RDF [17] representation captures explicit semantics of LOD data in an unambiguous way.

While MetaMap has been widely adopted by the research community, it still suffers from several issues. First, UMLS does not have a satisfactory updating mechanism for its domain entities. Compared to the rapidly increasing LOD sources, this individual metathesaurus has a relatively lower update frequency and scale increment speed, which makes MetaMap encounter a data flexibility problem. Second, it supports natural language processing on English text only and, therefore, is not useful for other languages. Third, it uses a rule-based inference for word sense disambiguation, which always leads to local optimization, producing in many cases inaccurate results.

In this paper, we propose a knowledge-incentive approach for entity annotation in biomedicine based on LOD. Its main features are: (a) By exploiting the openness of LOD, it is data flexible; (b) By incorporating new related LOD sources, it achieves a much better performance; (c) Based on the multilingual nature of LOD, it is language independent; and (d) By using the rich semantic relationships among entities that are available in LOD, it efficiently addresses the disambiguation problem producing much more accurate results. Instead of rule-based inference, we adapt the Collective Annotation [15] process to find the best matches for all text references globally. Based on this approach we developed MeDetect, a prototype system for domain entity extraction for biomedical references.

[1] Statistical information of the Web data sources: http://richard.cyganiak.de/2007/10/lod/

[2] For example, most entities in DBpedia, one of the 295 linked data sources, are well described in more than 10 languages.

978-1-4799-2902-3/13 $31.00 © 2013 IEEE
DOI 10.1109/WI-IAT.2013.34

The experimental results verify the effectiveness and efficiency of our approach.

Overall, the main contributions of the paper can be summarized as follows:

1. We propose a novel knowledge-incentive approach based on LOD for entity annotation in biomedicine. This approach has advantages in data flexibility, language independence, and semantic relationship enrichment, which makes it more convenient and informative for further applications.
2. We propose the use of collective annotation leveraged by LOD information to conduct entity filtering and disambiguation.
3. We develop MeDetect to implement the proposed approach, and conduct experiments to verify its effectiveness and efficiency.

The remainder of the paper is organized as follows. In Section 2 we discuss related work on named entity recognition and linked data consumption. We describe our methodology in Section 3. In Section 4 we present the design and results of the experiments, and in Section 5 we discuss the results. Finally, we conclude the paper and discuss future work in Section 6.[3]

II. Related Work

A. Named Entity Recognition

The concept of named entity, introduced in 1996 [12], is now widely used in natural language processing and the Semantic Web. For both unstructured natural language content and semi-structured Web pages or references, it is essential to capture metadata related to the content of these pages, including person names, organizations, locations, and other relevant concepts, making named entity recognition (NER) a popular topic in both academia and industry.

Most early work on named entity recognition was based on proper names, always relying on rule-based approaches [20], such as linguistic grammar-based techniques in the target language [24, 23]. Such approaches usually achieve high precision but low recall.

Machine Learning-based approaches have been proposed to overcome the problem of low recall of rule-based approaches, while they further aim at recognizing some unknown entities. Supervised learning used in NER refers to learning the features of labeled named entities in the training data and identifying unknown entities in test data. The most prominent supervised learning models in NER include Support Vector Machine [25], Hidden Markov Models [8], and Conditional Random Fields [18]. Instead of requiring a large set of training cases like supervised learning approaches, semi-supervised approaches [21, 10] start from a small set of "seed" cases and expand the named entities based on the supervised model trained on the current training set.

Recent studies on social networks propose using additional types of information for named entity recognition. For example, the authors of [16, 11] use data that refers to the Twitter users' social behavior to improve the named entity recognition performance in Tweets.

B. Linked Open Data Consumption

Since the Linked Open Data (LOD) [9] Initiative was launched by W3C in 2007, there have been 295 data sources published in the LOD Cloud. Besides general encyclopedia-like sources, such as DBpedia and Freebase, they cover a variety of more specific fields including Geography (GeoNames, LinkedGeoData), academia (DBLP, OAI), music (MusicBrainz, BBCMusic), transportation (transportdata.gov.org.uk) and Biomedical Informatics (Medicare, PubMed). In general, each data source in the LOD Cloud can be regarded as a domain knowledge base, which is highly valuable for data mining [22], information retrieval [27], knowledge reasoning and representation [19] in its domain. The number of applications consuming linked data for these kinds of task has become very large, as also evidenced by the Consuming LOD workshop, which was held at the International Semantic Web Conference in 2009.

In this work, we annotate biomedical references with the entities from biomedical LOD sources, leveraging the link information of LOD entities to improve the annotation performance. The annotated entities belong to LOD, so there is much potential for further tasks such as knowledge mining, reference categorization and others, etc.

III. Methodology

We developed the entity glossary of MeDetect using biomedical data from LOD, which assigns one or more Uniform Resource Identifiers (URIs) to each entity name.

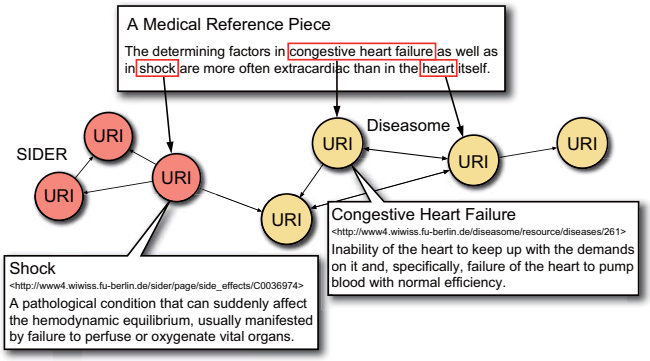

Fig. 1. An Example of MeDetect Entity Annotation

For an input biomedical reference, MeDetect first parses each sentence and extracts the matched entity names using its entity glossary. Then entity filtering and disambiguation is conducted to get the more accurate entity annotation for the

[3] This full paper is an extended work based on our previous poster paper [26], with full system/model description, further experimental evaluation, and the final case study.

input reference. Figure 1 shows an example of MeDetect entity annotation for a piece of biomedical reference.

The overall design of MeDetect is shown in Figure 2. The off-line processing part is in pink while the on-line part is in yellow. The system consists of three components, which correspond to three steps of domain entity annotation for biomedical references. First, Data Preparation extracts useful entity information from selected linked open data sources, and generates the glossary and entity relation data. This step is applied off-line. Second, Entity Name Recognition finds entities from the glossary based on the match between their names and the input biomedical reference content. These entities with their URIs are regarded as candidates for the final output. Third, Entity Filtering and Disambiguation removes irrelevant entities and disambiguates each entity to select the best URI for each entity name. It leverages collective annotation [15], which takes into account both node importance and inter-node coherence. The detailed algorithms and implementations of these components are elaborated in the following sections.

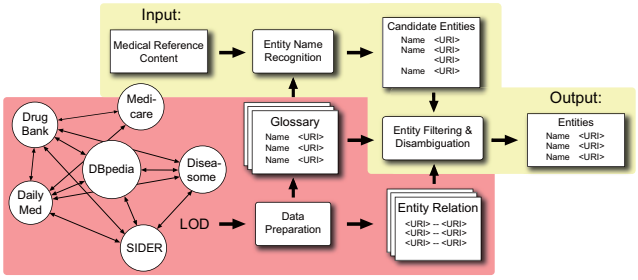

Fig. 2. The architecture of MeDetect

A. Data Preparation

This step is off-line. It generates two thesauruses for the on-line process of MeDetect. The first one is an entity glossary, which can be regarded as the dictionary for annotation. The second thesaurus contains entity relation data, which is used in entity filtering and disambiguation. All the entity information of MeDetect comes from Linked Open Data, which consists of different data sources with links between their instances. Different data sources often use different schemas. The data preparation component aims to analyze the schemas, create data extraction rules, and finally integrate this data.

The main data source that we use is Linked Open Drug Data (LODD). The data in LODD refers to drug effects and clinical trials. LODD also focuses on linking various drug data sources to handle some application problems. We use the five basic data sources of LODD: DrugBank, Diseasome, Medicare, SIDER, and DailyMed. Each data source consists of a set of RDF triples following the corresponding schema. With the analysis of each schema, we can apply appropriate rules for entity information extraction. For example, List 1 shows a small part of the RDF schema in DrugBank. By selecting RDF triples with descriptive predicates such as `rdf-schema#label`, `drugbank/synonym`, or `drugbank/genericName`, we can construct the glossary in the EntityName-URI format, as shown in List 2.

```
<http://www4.wiwiss.fuberlin.de/drugbank/resource/
drugs/DB00001>
<http://www.w3.org/2000/01/rdf-schema#label>
"Lepirudin" .

<http://www4.wiwiss.fuberlin.de/drugbank/resource/
drugs/DB01073>
<http://www4.wiwiss.fuberlin.de/drugbank/resource/
drugbank/synonym>
"FAMP" .
```

List 1. Samples of selected useful RDF triples in DrugBank

```
"Lepirudin"
<http://www4.wiwiss.fuberlin.de/drugbank/resource/
drugs/DB00001>

"FAMP"
<http://www4.wiwiss.fuberlin.de/drugbank/resource/
drugs/DB01073>
```

List 2. The corresponding part of glossary to the RDF triples in List 1

By processing the five LODD sources, we obtain 85,631 EntityName-URI pairs in the glossary. As shown in Lists 1 and 2, the entities in LODD are always specific drug ingredient names, clinical trials, diseases, and pharmaceutical companies, all very specific to the field of biomedicine. However, in biomedical references, there is also some less domain-specific terminology. Thus the coverage of entity names from LODD is not broad enough for entity annotation from biomedical references. To address this issue, some less specific data sources should be added in the glossary.

DBpedia [7] is a well-known entity source, which covers various fields, including biomedicine. In its 3.7 release in November 2011, there are already 3.64 million entities described in English. Considering its broad coverage, we added a subset of DBpedia entities filtering out relevant entities. As it is also shown in Figure 2, there are `owl:sameAs` links connecting LODD sources to DBpedia, which indicates the existence of some ground-truth biomedical entities in DBpedia. Using these links, we can obtain 6,718 biomedical entities from DBpedia. This is a relatively small set of high quality entities, called the seed set. We need to expand this set in order to get more relevant entities with a small loss of quality.

We propose a two-stage approach to address this problem. In the first stage, we expand the entity set by adding additional entities that share a common category with any entities in the original seed set, which leads to 43,349 more entities. We obtain these entities with a coarser-grained process, and as a result they also contain noise data. Therefore, in the second stage, we conduct an elaborate bi-classification algorithm on this expanded entity set to judge whether each entity is a biomedical related entity. Specifically, we use a Support Vector Machine [25] (SVM) as the bi-classification algorithm. For the training data, we sample 4/5 of the seed set entities and an equal number of unlabeled entities as the labeled positive and

negative data respectively[4]. The remaining 1/5 of the labeled data is taken as validation data to tune the parameters of the SVM. The features of each entity come from its InfoBox data. After the training phase, the SVM predicts whether each entity in the expanded set from stage 1 is a biomedical entity. This gives 93% accuracy on validation data and 88% accuracy in sampled test data. With this step, we obtain 12,753 more entities with 88% accuracy. Eventually, the glossary size has increased to 105,102.

Besides entity glossary construction, in MeDetect we also need inter-entity relation information to support the step of entity filtering and disambiguation. Since the entities in the glossary are extracted from RDF triples, there is naturally relation data between these entities or properties of these entities. List 3 gives two examples of entity relations and property data.

```
<http://www4.wiwiss.fuberlin.de/dailymed/resource/
organization/Parke-Davis>
<http://www.w3.org/2002/07/owl#sameAs>
<http://data.linkedct.org/resource/agency/106> .

<http://www4.wiwiss.fuberlin.de/dailymed/resource/
drugs/3037>
<http://www4.wiwiss.fuberlin.de/dailymed/resource/
dailymed/name>
"Hydromet" .
```

List 3. Examples of entity related and property data in the format of RDF triples in DailyMed

For these kinds of entity data, we extract the subject and object entities in each triple to make a pair and store in the EntityURI-EntityURI format. Also this relation data is indexed by *Lucene* [13], which is an open-source package for information retrieval.

To sum up, the off-line data preparation step produces the entity glossary and relation data for the on-line process of MeDetect. The flexibility and openness of LOD also makes MeDetect flexible and open.

B. Entity Name Recognition

With the biomedical entity glossary, the on-line entity name recognition step provides the syntactic match between the entity name in the glossary and the content of input biomedical references. This is based on syntactic matching with some lexical processing (word normalization). The recognized entities with their URIs are passed to the next step as candidate entities to be further filtered.

First, we parse each sentence and apply a traditional lexical process to normalize each word. We also filter the stop words. Then we look for matches between the words or phrases and the entity names in the glossary. Finally, for each matched entity name, we retrieve all its indexed URIs as a package to pass to the next step. Each entity name is definitely assigned no less than one URI. In the entity name match process, there is a problem of entity name overlap. For example (also see Figure 3), consider the following input sentence: "*The determining factors in congestive heart failure as well as in shock are more often extracardiac than in the heart itself*". For this sentence, "congestive heart failure", "heart failure" and "heart" all have their own correspondence in the entity glossary. One possible solution is to recognize and extract all three entity names as the candidate entities. Another solution is to just extract the longest-name entity "congestive heart failure" as candidate entity. Based on an analysis of entity names in biomedicine, we chose the second solution. One reason is that for two entity name strings of entity A (such as "congestive heart failure") and entity B (such as "heart failure"), where the name of B is a substring of that of A, it always appears that A is medically more specific than B. Another reason is that by further entity extraction on the name string of A, we can obtain B. In addition, A and B are usually interlinked in knowledge bases. Thus by just extracting the longest-name entities, we can capture all the information of the entities contained in the input text.

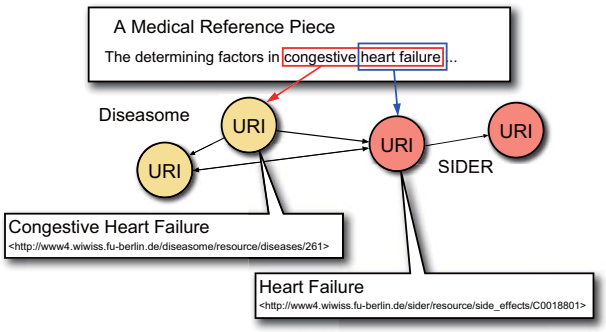

Fig. 3. The problem of entity name overlap. In our work, we choose to match the longest-name entity. The shorter-name entities can be extracted by further annotation or using linked data

The same approach has also been widely used in the application of Web page annotation [28]. An alternative approach for entity matching is *indirect matching*. For example, for a phrase of the form "… tuberculous hydrothorax …", one possible candidate entity is "tuberculosis". Here the attribute "tuberculous" refers to the entity "tuberculosis". However, there are some difficulties in transferring all entity-related attributes to their corresponding entity names. Therefore, we do not implement this part in our current work and leave it as future work.

C. Entity Filtering and Disambiguation

The last but most important on-line step of MeDetect is entity filtering and disambiguation. This step addresses the following two issues: (a) Some words or phrases should not be regarded as annotations of the input reference, even if they co-occur in the input reference and the glossary, such as the noise entities in the glossary or the ones with URIs in other biomedical topics; and (b) Words and phrases and entity

[4] Since there is only very small part of entities in DBpedia belonging to the biomedical field, regarding the randomly chosen unlabeled entities as non-biomedical entities is of low noise.

names may be ambiguous. For example, "*cold*" can either mean "*low temperature*" or "*a common illness*", depending on the context.

Collective Annotation. In MeDetect, we implement and revise the most recent work [15] on Web page annotation, called *Collective Annotation*. For each input reference, let y denote the vector of URIs for all candidate entities. For each candidate entity i, y_i denotes one URI retrieved from the entity glossary for i. N/A is among the potential values assigned to y_i, which means that there is no relevant URI for candidate entity i. This annotation approach not only detects the importance of each entity i (with its URI y_i) for the input reference, but also, more importantly, filters some irrelevant entities based on the inter-entity relationship $r(y_i, y_j)$. To sum up, as shown in Eq. (1), there should be two functions in collective annotation: the single entity importance function $S(y_i)$ and the entity clique coherence function $C(y)$.

$$\begin{aligned} \arg\max_y \Pr(y) &= \arg\max_y C(y) \prod_i S(y_i) \\ &= \arg\max_y \log C(y) + \sum_i \log S(y_i) \\ &= \arg\max_y \log \sum_{i \neq j} r(y_i, y_j) + \sum_i \log S(y_i) \end{aligned} \quad (1)$$

The single entity importance function $S(y_i)$ estimates the relevance between an entity and the input reference, based on their syntactic and semantic similarity, such as the logistic regression model or category-based matching. Here, the entity description information in its URI can be utilized to match the input reference. If the size of the input reference is relatively large, the relevance is then estimated between the entity and its context, which is usually a suitable text window around the place where the entity name occurs. In our scenario, the importance of each entity is based on its origin score (such as 0.88 for the entities expanded from SVM) and the semantic match between entity URIs and the context.

The entity clique coherence function $C(y)$ measures the topic similarity or consistency of the whole set of Entity URIs so as to filter out noise entities and cope with the ambiguities of entities. For example, if a candidate entity has no relationship or common topic with others, it is likely that this candidate is noise. Also if a candidate entity name is assigned more than one URI (y_i), the entity pair coherence function $r(y_i, y_j)$ will calculate the coherence of each of these URI with the ones of other entities and choose the most coherent one as the final URI of this entity name. Thus the problem of ambiguity is handled. In MeDetect, we use a LOD neighborhood overlap calculation [29] to implement entity pair coherence function $r(y_i, y_j)$, as shown in Equation 2:

$$r(y_i, y_j) = \frac{|N(y_i) \cap N(y_j)|}{\max(|N(y_i)|, |N(y_j)|)}, \quad (2)$$

where $N(y_i)$ denotes the set of neighbor entities to y_i in the LOD. Figure 4 gives an illustration of LOD neighborhood overlap $r(y_i, y_j)$.

With the joint optimization on the single entity importance function and entity pair coherence function, MeDetect filters out noise entities and disambiguates each entity name with more than one URIs. Finally, MeDetect outputs the selected entity names with their unique URI as the domain entity annotation for the input biomedical reference.

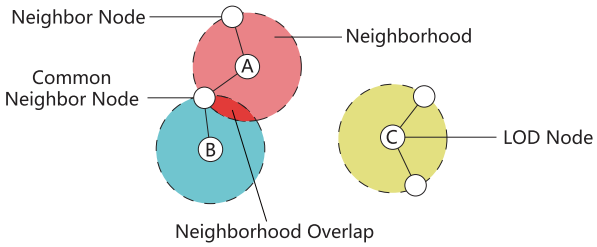

Fig. 4. LOD neighborhood overlap calculation in collective annotation of MeDetect

IV. EXPERIMENTS

We evaluated the effectiveness and efficiency of MeDetect with an experimental study. We test with the biomedical paper abstract texts and compare with several biomedical named entity annotation systems. The experimental results show great effectiveness and efficiency of MeDetect. Finally, we provide a case study to illustrate some results of MeDetect.

A. Setup of The Experiments

Linked Open Data. As described in Section 3.1, the entity glossary and entity relations are derived from the linked open data sources. We collect five sources from Linked Open Drug Data and use a supervised learning approach to extract the biomedical related entities in DBpedia. The resulting numbers of pairs of various types are in Table 1.

LODD Source	Entity-URI Pairs	Entity Relation Pairs
DailyMed	8,463	164,276
DBpedia	19,471	130,986
Diseasome	13,979	91,182
DrugBank	44,760	766,920
Medicare	13,477	44,500
SIDER	4,952	193,249
Total	105,102	1,391,113

Table 1. Contributions to entity glossary and entity relation from different data sources

Test Biomedical References In our experiment, 120 paper abstracts with different biomedical topics are randomly selected from PubMed as the test biomedical references. The average word length of these references is about 80. For each input reference, MeDetect outputs all the confident entities assigned single URI and the other biomedical name entity annotation systems output the confident entities with the assigned sense.

Evaluation Measures. For the annotation quality evaluation, we invited three experts with biomedical or computer science background to score the output entities for each paper. For each entity, if the expert believes it is accurate as an annotation for the input reference, he/she scores the entity with 1; otherwise with 0. To avoid individual bias, each output entity is evaluated by at least two experts[5]. The judgment on each output entity should take its URI information or sense into consideration to avoid the ambiguities of each entity name.

Comparative Evalaution. In our experiments we compare MeDetect with two other related systems: MetaMap and LingPipe [1]. MetaMap is an entity extraction system with a large glossary built on UMLS, an elaborate lexical processor, and a disambiguation component. Entities are recognized by a series of language rules. More specifically, for best effectiveness and efficiency, we choose its semantic mode (with word sense disambiguation). LingPipe is a model-based system for processing text using computational linguistics. For both MetaMap and LingPipe, their output entities are not given specific URIs but short entity descriptions (which can also be regarded as phrase senses), e.g. Shock [Pathologic Function] and Burn injury [Injury or Poisoning]. Finally, we turn off MeDetect's entity filtering and disambiguation component to evaluate the effectiveness of its collective annotation algorithm, calling this approach MeDetect\FD.

B.

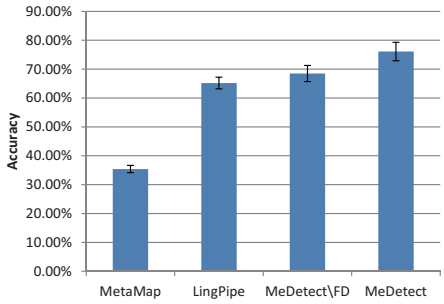

Fig. 5. Overall accuracy comparison

C. Overall Accuracy Comparison

We compare the accuracy of the four systems in Figure 5 and more details are in Table 2. From Table 2 we can make the following observations. (i) Without the entity filtering and disambiguation component, MeDetect\FD provides a lower accuracy, despite a few more correctly recognized entities. Based on this we conclude that collective annotation can improve the annotation accuracy, even though some correctly recognized entities are missed. (ii) The average accuracy of MeDetect is much higher than MetaMap or LingPipe, which indicates the effectiveness of utilization of biomedical related linked data. (iii) Although MetaMap and LingPipe output more entity annotations, the final number of correctly recognized entities is not much higher, which indicates that there is much noise in the output of MetaMap and LingPipe. Overall, we observe that MeDetect can provide sufficiently high quality entity annotations for biomedical references. In addition, with a URI assigned to each entity, MeDetect is ready for further applications, such as biomedical triple extraction.

Compared systems	#Test references	#Output entities	#Correc-ted	Average accuracy
MeDetect	120	598	455	**76.1%**
MeDetect\FD	120	683	468	68.5%
MetaMap	120	1,062	412	35.4%
LingPipe	120	782	510	65.2%

Table 2. Output entities, correctly recognized entities, and accuracy of each system

D. Efficiency Comparison

Besides effectiveness, efficiency is also a key factor for every annotation system. We have a running time comparison among MeDetect, MetaMap, and LingPipe. Table 3 shows the running time of these three systems and Figure 6 illustrates this comparison. We observe that MeDetect is much more efficient than MetaMap (about 30 times faster) and LingPipe (about 3 times faster). Actually, LingPipe already has an on-line service. The average running time of 20.2 milliseconds suggests that MeDetect can also be deployed on-line.

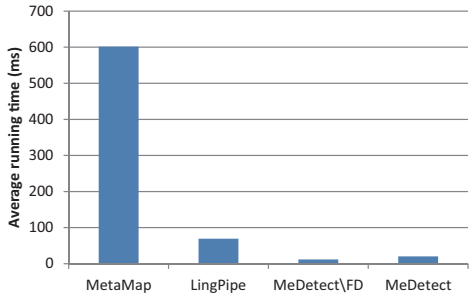

Fig. 6. Running time comparison

Compared sytems	#Documents	Total running time (ms)	Average running time
MeDetect	120	2,421	20.2
MeDetect\FD	120	1,422	**11.9**
MetaMap	120	72,162	601.4
LingPipe	120	8,320	69.3

Table 3. Total and average running time of each system

[5] The contradiction between experts is shown in the form of error bar in Figure 5.

E. *Multiple LOD Sources Contribution*

To illustrate the usefulness of multiple LOD sources, we compare the performance of MeDetect with different sources: DBpedia, LODD, and the combination of both. The results are shown in Table 4. We can see the average accuracies are almost the same. Adding more related LOD sources will improve the performance of MeDetect. Particularly, the average number of correctly recognized entities increases from 2.44 to 3.79 with almost no accuracy decrease when adding LODD sources. On the one hand, more LOD sources will enrich the entity glossary with more entity names and more URIs for each entity. This is also the reason why the number of output entities does not equal the sum of the first two. On the other hand, more URIs will improve the performance of collective annotation, the component that does entity filtering and disambiguation.

Compared Data Sources	#Test refs	#Output entities	#Corrected entities	Avg accuracy	Avg corr. recognized entities
DBpedia	120	394	293	74.4%	2.44
LODD	120	330	238	72.1%	1.98
DBpedia + LODD	120	598	455	**76.1%**	**3.79**

Table 4. Performance comparison of MeDetect with different combinations of LOD sources

To further analyze the contribution of each LOD source, we measure the entity URI distributions on each LOD source used in MeDetect after the steps of entity name recognition (Section 3.2) and entity filtering & disambiguition (Section 3.3). The former distribution indicates the domain entity coverage of each LOD source on the test references, while the latter shows the final contribution of each LOD source. The results are shown in Figure 7.

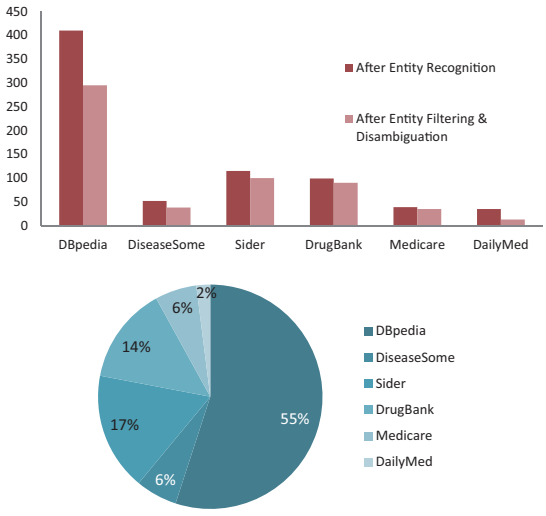

Fig. 7. Annotated entity distributions on different LOD sources

The upper subfigure of Figure 7 presents the annotated entity distributions on each LOD source after two steps of MeDetect, while the lower one gives their proportion comparison from the output (after entity filering and disambiguition). It is obvious that DBpedia entities take a large proportion after both steps. On the other hand, the entities from most LODD sources, except for DailyMed, have a high survival rate in the step of entity filering and disambiguition. This is because the LODD sources always have a high correlation with each other and there are a number of links between their entities, which gives advantages in the collective annotion. Overall, DBpedia biomedical entities accout for a large part of the results of MeDetect, while the other domain-specific biomedical LOD sources contribute entities of high quality. Thus expanding the DBpedia biomedical entity set and including more professional biomedical LOD sources will lead to higher recall and accuracy of MeDetect.

F. *Case Study*

Finally, we end this section with a case study of MeDetect. In Table 5, the input paper abstract is in the upper part and the entity annotations with their frequency in this abstract and URIs are listed below. The following observations verify our experimental results. (i) The entities are all about biomedical topics and relevant to the input paper abstract. (ii) The recall is also high. All the biomedical entities in the input paper abstract are extracted except for the acronym "*CEA*". (iii) The output entities are given URIs from DBpedia, Diseasome, SIDER, and DrugBank, which indicates that multiple LOD sources are important and useful. (iv) More LOD sources are needed for further improvement. Also some acronym mappings can be added in the entity glossary.

Input Paper Abstract:

Chronic kidney disease is an important risk factor for development and progression of atherosclerosis. The objective of the current study was to investigate the contribution of moderate kidney failure to cardiovascular (CV) mortality and morbidity after carotid endarterectomy (CEA). In addition, we investigated which proportion received optimal medical treatment or underwent diagnostic workup of the kidneys prior to CEA.

Entity Name	Entity URI
Carotid Endarterectomy	<http://www4.wiwiss.fu-belin.de/sider/resource/side_effects/C0014099>
Cardiovascular	<http://www4.wiwiss.fu-berlin.de/diseasome/resource/diseaseClass/Cardiovascular>
Chronic Kidney Disease	<http://www4.wiwiss.fu-belin.de/sider/resource/side_effects/C0022661>
Atherosclerosis	<http://www4.wiwiss.fu-belin.de/sider/resource/side_effects/C0004153>
Kidney Failure	<http://www4.wiwiss.fu-belin.de/sider/resource/side_effects/C0035078>
Development	<http://www4.wiwiss.fu-berlin.de/drugbank/resource/drugbank/goClassificationProcess>
Medical	<http://dbpedia.org/resource/Medical>

Table. 5. A Case Study of MeDetect

V. DISCUSSION

The experimental results verify the high accuracy of MeDetect where a URI is assigned to each annotated entity. We also observe that by using more related LOD sources, we can improve the performance of MeDetect. So far, we have utilized five LODD sources and DBpedia. More biomedical domain related LOD sources, such as UMLS and UniProt data sources in Linked Life Data, could be used to improve accuracy and recall for domain entity annotation. We could also add biomedical entities from general domain LOD sources, such as Freebase and Yago, in the same way that we did with DBpedia.

Since the number of entities with URIs is still smaller than the number without, one challenge for MeDetect is to make use of unlabeled entities to improve the labeled entity annotation. For example, we could match the words or phrases in the biomedical dictionary to the labels of entities on LOD sources, such as DBpedia and Freebase, to expand the entity glossary.

Another challenge will arise with the increasing size of LOD source size of our glossary. In this case, the entity pair coherence function in collective annotation will possibly have lower efficiency. We plan to use an optimized search index to store the neighborhood information for each node to deal with this forthcoming problem.

VI. CONCLUSION

This paper describes a novel knowledge-incentive approach based on LOD for domain entity annotation in biomedical references. This approach has data flexibility, language independence, and semantic relationship enrichment, which make it more suitable and informative for further applications. Based on this approach, we implemented a prototype system, MeDetect. Its three key components are: data preparation, entity name recognition, and entity filtering and disambiguation. The results from the experiments that we conducted indicate that MeDetect demonstrates high annotation accuracy and good efficiency; it also has data flexibility for adding more LOD sources. In future work, we will enrich the entity glossary of MeDetect by adding more LOD sources, indirect entity match, and acronym mappings. More importantly, MeDetect will be further utilized for triple extraction from biomedical references.

REFERENCES

[1] Alias, I. LingPipe 4.1.0. http://alias-i.com/lingpipe (2008)

[2] Linked Life Data. http://linkedlifedata.com

[3] Linked Open Drug Data. http://www.w3.org/wiki/HCLSIG/LODD

[4] Aronson, A.R.: Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program. In: Proc. of AMIA. (2001)

[5] Aronson, A.R., Lan, F.M.: An Overview of MetaMap: Historical Perspective and Recent Advances. In: Proc. of JAMIA. (2010)

[6] Aronson, A.R., Mork, J.G., Gay, C.W., Humphrey, S.M., Rogers, W.J.: The NLM Indexing Initiative's Medical Text Indexer. In: Proc. of Medinfo. (2004)

[7] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives Z.: DBpedia: A Nucleus for a Web of Open Data. In: The Semantic Web. (2008)

[8] Bikel, D. M., Miller, S., Schwartz, R., Weischedel, R. Nymble.: A High-Performance Learning Name-finder. In: Proc. of CANLP. (1997)

[9] Christian, B., Tom, H., Tim, B.L.: Linked Data-The Story So Far. In: Proc. of IJSWIS. (2009)

[10] Collins, M.: Ranking Algorithms for Named–Entity Extraction: Boosting and the Voted Perceptron. In: Proc. of ACL. (2002)

[11] Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J., Dredze, M.: Annotating Named Entities in Twitter Data with Crowdsourcing. In: Proc. of the NAACL HLT Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. (2010)

[12] Grishman, R., Sundheim, B.: Message Understanding Conference - 6: A Brief History. In: Proc. of International Conference on Computational Linguistics. (1996)

[13] Hatcher, E., Gospodnetic, O.: Lucene in Action. In: Manning Publications. (2004)

[14] Jentzsch, A., Zhao, J., Hassanzadeh, O., Cheung, K.H., Samwald, M., Andersson, B.: Linking Open Drug Data. In: Proc. of Triplification Challenge. (2009)

[15] Kulkarni, S., Singh, A., Ramakrishnan, G., Chakrabarti, S.: Collective Annotation of Wikipedia Entities in Web Text. In: Proc. of SigKDD. (2010)

[16] Liu, X., Zhang, S., Wei, F., Zhou, M.: Recognizing Named Entities in Tweets. In: Proc. of ACL. (2011)

[17] Manola, F., Miller, E.: RDF primer. In: W3C recommendation. (2004)

[18] McCallum, A., Li, W.: Early Results for Named Entity Recognition with Conditional Random Fields, Features Induction and Web-Enhanced Lexicons. In: Proc. of Conference on Computational Natural Language Learning. (2003)

[19] Moran, S.: Using Linked Data to Create A Typological Knowledge Base. In: Linked Data in Linguistics. (2012)

[20] Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. In: Lingvisticae Investigationes. (2007)

[21] Nadeau, D., Turney, P., Matwin, S.: Unsupervised Named Entity Recognition: Generating Gazetteers and Resolving Ambiguity. In: Proc. of CCAI. (2006)

[22] Passant, A. Dbrec—Music Recommendations using DBpedia. In: Proc. of ISWC. (2010)

[23] Piskorski, J.: Named-entity recognition for Polish with SProUT. In: Intelligent Media Technology for Communicative Intelligence. (2005)

[24] Shaalan, K., Raza, H.: Person name entity recognition for Arabic. In Proc. of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources. (2007)

[25] Suykens, J.A.K., Vandewalle, J.: Least Squares Support Vector Machine Classifiers. In: Neural Processing Letters. (1999)

[26] Tian, L., Zhang, W., Wang, H., Wu, C., Ni, Y., Cao, F., Yu, Y.: MeDetect: Domain Entity Annotation in Biomedical References Using Linked Open Data. In: Proc. of PD track of 17th International Semantic Web Conference. (2012)

[27] Vallet, D., Fernandez, M., Castells, P. An Ontology-based Information Retrieval Model. In: The Semantic Web: Research and Applications. (2005)

[28] Zhang, W., Wang, D., Xue, G.R., Zha, H.: Advertising Keywords Recommendation for Short-Text Web Pages Using Wikipedia. In: ACM TIST. (2012)

[29] Zhou, W., Wang, H., Chao, J., Zhang, W., Yu, Y.: LODDO: Using Linked Open Data Description Overlap to Measure Semantic Relatedness Between Named Entities. In: Proc. Of JIST. (2011)