UMLS to DBPedia Link Discovery Through Circular Resolution

John Cuzzola[1❄], MSc
jcuzzola@ryerson.ca

Ebrahim Bagheri[1], PhD
bagheri@ryerson.ca

Jelena Jovanovic[2], PhD
jeljov@gmail.com

❄ Corresponding author

1. Laboratory for Systems, Software and Semantics (LS3), Ryerson University, Ontario, Canada, http://ls3.rnet.ryerson.ca

2. Faculty of Organizational Sciences (FOS), University of Belgrade, Belgrade, Serbia, http://www.fon.bg.ac.rs/eng/

Abstract

Objective: The goal of this work is to map UMLS concepts to DBpedia resources using widely accepted ontology relations including skos:exactMatch, skos:closeMatch, and rdfs:seeAlso, as a result of which a complete mapping from UMLS[1] to DBpedia[2] is made publicly available that includes 221,690 skos:exactMatch, 26,276 skos:closeMatch, and 6,784,322 rdfs:seeAlso mappings.

Materials and Methods: We propose a method called *circular resolution* that utilizes a combination of semantic annotators to map UMLS concepts to DBpedia resources. A set of annotators annotate definitions of UMLS concepts returning DBpedia resources while another set performs annotation on DBpedia resource abstracts returning UMLS concepts. Our pipeline aligns these two sets of annotations to determine appropriate mappings from UMLS to DBpedia.

Results: We evaluate our proposed method using structured data from the Wikidata knowledge base as the ground truth, which consists of 4,899 already existing UMLS to DBpedia mappings. Our results show an 83% recall with 77% precision-at-one (P@1) in mapping UMLS concepts to DBpedia resources on this testing set.

Conclusion: The proposed circular resolution method is a simple yet effective technique for linking UMLS concepts to DBpedia resources. Experiments using Wikidata-based ground truth reveal a high mapping accuracy. In addition to the complete UMLS mapping downloadable in n-triple format, we provide an online browser and a RESTful service to explore the mappings.

[1] UMLS 2016AA
[2] DBpedia 2015-10

INTRODUCTION

DBpedia is a crowd-sourced community project for extracting structured, multilingual information from Wikipedia to be made freely available on the Web in machine intelligible format based on Semantic Web standards [1]. It is the central component and the main interlinking hub in the Linked Open Data (LOD)[3] cloud, a network of open structured datasets published on the Web according to the Linked Data principles [2]. LOD consists of several billion interlinked data points and covers a wide variety of domains such as geography, government, life sciences, media, social networking, scientific publications, to name a few. Whereas biomedical datasets constitute a large portion of the LOD cloud[4], and several of these datasets are connected to DBpedia, the complete integration of the UMLS Metathesaurus is still missing. If available, a mapping between DBpedia resources and UMLS concepts could provide several benefits to the biomedical community.

The work presented in this paper aims at providing a bridge connecting UMLS to DBpedia, in a manner that is both efficient, i.e., fully automated, and effective, i.e., highly accurate. In particular, the contribution of the presented work is twofold:

1. We introduce a method of automated link discovery between equivalent, near-equivalent, and related concepts originating from two large-scale knowledge bases (KBs), namely, UMLS Metathesaurus and DBpedia;
2. We release a publicly available complete mapping set between UMLS and DBpedia that can facilitate the integration of many biomedical and medical KBs, through UMLS, to the Linked Open Data cloud.

[3] https://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData

[4] As obvious from the LOD cloud diagram: http://lod-cloud.net/

BACKGROUND

Significance

The significance of the UMLS to DBpedia mapping presented in this paper is multifold:

- Sophisticated text/data mining tasks depend on the availability of KBs built from diverse sources [3]. Wikipedia contains large amounts of scientific and medical data, and thus has been recognized as highly useful for setting up initial KB for biomedical projects [4]. It has also proven useful for estimating semantic similarity of gene pairs [5]. In particular, Dessi & Atzori demonstrated that Wikipedia's 10K+ articles about human genes allow for highly accurate assessment of gene similarity and detection of functional groups of genes. The machine-readable version of Wikipedia, DBpedia, is also a highly rich knowledge source with the additional advantage of enabling automated and machine-intelligible access to the knowledge it contains. For instance, Yamamoto et al. used DBpedia to automatically extend a life science database of abbreviations and their long forms (LFs) with additional descriptions of the LFs, thus enabling users to more easily select the correct LF for a particular abbreviation [6].
- As the central hub in the LOD cloud, DBpedia offers connection to numerous biomedical and other related datasets and KBs. Based on the latest statistics, DBpedia is connected to other LOD datasets through an estimated 50 million links. This indicates that DBpedia can serve as a hub for accessing diverse types of data for building rich KBs.
- Based on search engine ranking and page view statistics, the English Wikipedia is a prominent source of online health information [7]. DBpedia has the potential to be even more useful, as it provides grounds for building advanced applications that not only facilitate information search and retrieval, but also act proactively, e.g. , applications that recommend resources a user has not explicitly asked for but might benefit from (see, e.g., [8]). In addition, it can be used to further advance the current approaches for assessing the trustworthiness of online health information. For example, Park et al. [9] demonstrated that online health-related content annotated with Wikipedia

concepts can be effectively used for building page-level and site-level classifiers aimed at differentiating between trustworthy and suspicious sites. It is reasonable to expect that the performance of such classifiers could be further improved if the Wikipedia concepts, identified in Web pages, are mapped to the corresponding UMLS concepts, thus allowing for a more precise semantic representation of health-related content of Web pages.

- Finally, a UMLS to DBpedia mapping can be relevant for bridging the gap between health-related jargon used by professionals and that used by the general public [10]. For instance, having examined ten large online question corpora, Roberts and Demner-Fushman found that consumers, i.e., the general public, used significantly less medical terms than medical professionals [11]. Likewise, consumers' questions were found to be closer to an open-domain language model, built on newswire and Wikipedia, than to a medical model, built on a sample from PubMed Central. This was further confirmed by Mrabet et al. who demonstrated that combining an open-domain KB (i.e. DBpedia) with a biomedical KB (i.e. UMLS) could lead to a substantial improvement in identifying the main topics of consumer health questions [12]. These findings suggest that DBpedia could be more suitable for semantic annotation, i.e., entity linking, of consumer questions, whereas UMLS would be more suitable for questions/answers coming from medical professionals; therefore, a mapping between UMLS and DBpedia can facilitate automated matching between (annotated) customers' questions and medical professionals' answers. In addition, it can be used to further improve the discovery and retrieval performance of systems for search and exploration of online content related to health and life sciences, such as DeepLife [13]. DeepLife's knowledge base covers a wide spectrum of biomedical entities, originating from UMLS and KnowLife [14], thus covering the needs and terminology of health and life science professionals. If extended with DBpedia/Wikipedia entities, through the proposed mapping, it would be better able to match search requests by the general public.

There have already been work within the biomedical and healthcare domains that employ open instance mapping platforms, such as Silk [15] and LIMES [16] to map across medical terminologies. For instance, Tilahun et al. used Silk to automatically link HIV-related data elements with data elements from Bio2RD, and LinkedCT [17]. In [18], Silk was used to map concepts between biomedical entities to help discover the side-effects of using thiazolinedione classed drugs such as Rosiglitazone. In [19], Silk linked proteomic, disease, and treatment data, to health records to find candidate patients for active clinical trials. Similarly, The Cancer Genome Atlas (TCGA)[5] used LIMES to build a massive, publicly available, 30 billion triple datastore of genetic genome mutations to advance discoveries against this disease [20]. There have also been work that have performed terminology mapping without using open mapping platforms. For example, Lee et al have used heuristics for mapping laboratory terminology to Logical Observation Identifiers Names and Codes (LOINC) [21]. Likewise, Kahn [22] has used semi-automated string matching to map Orphanet Rare Disease Ontology (ORDO) terms to the terms in the Radiology Gamuts Ontology (RGO). However, to the best of our knowledge, there has been no prior work that attempted to systematically map UMLS concepts to concepts from the widely used DBpedia knowledge base thus facilitating the integration of UMLS with the Linked Open Data cloud.

Ontological representation of equality relations

When formally expressing links between two knowledge bases, the most common relation is "equal-to" [23], often asserted using the predicate sameAs in the *Web Ontology Language* (OWL)[6], or by exactMatch in the *Simple Knowledge Organization System* (SKOS)[7]. The primary difference is owl:sameAs represents true equivalence in that every property of concept x is in the ontology of y and vice versa, whereas skos:exactMatch asserts that resource x is an exact match to resource y when both x and y can be used interchangeably for a wide range of information retrieval tasks. The predicate skos:closeMatch is similar to skos:exactMatch but does not necessarily preserve *transitivity*.

[5] https://cancergenome.nih.gov/

[6] https://www.w3.org/OWL

[7] https://www.w3.org/2004/02/skos/

Consequently, we intentionally avoid making the assertion of owl:sameAs because of strict equivalence requirements opting for skos:exactMatch/closeMatch as better choices given the published W3C standards. Furthermore, our method also considers the "seeAlso" property of the *Resource Description Framework Schema* (RDFS) that asserts that information about *x* might be available through resource *y*.

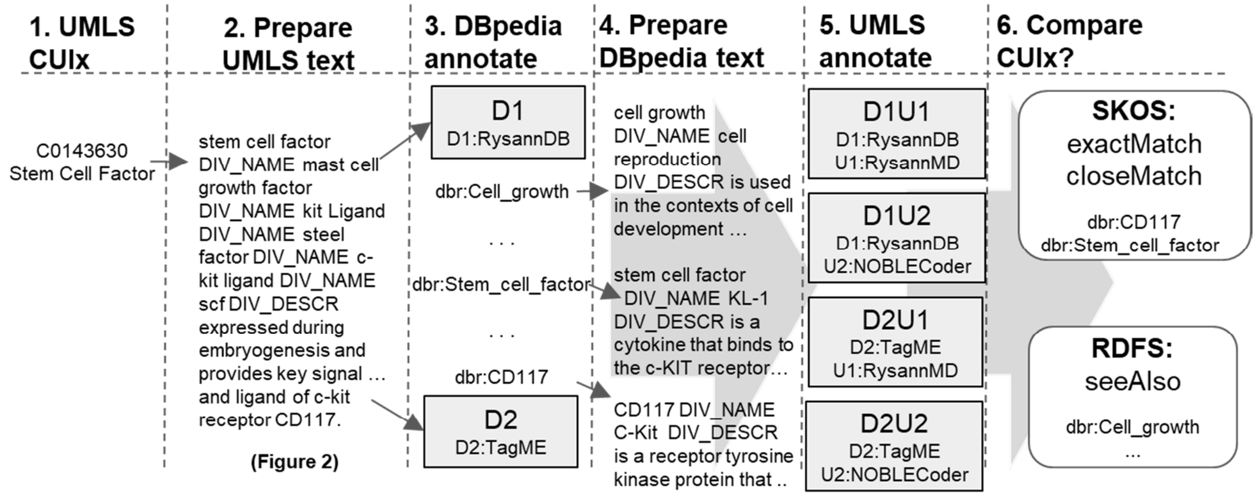
1. UMLS CUIx
2. Prepare UMLS text
3. DBpedia annotate
4. Prepare DBpedia text
5. UMLS annotate
6. Compare CUIx?
C0143630 Stem Cell Factor
stem cell factor DIV_NAME mast cell growth factor DIV_NAME kit Ligand DIV_NAME steel factor DIV_NAME c-kit ligand DIV_NAME scf DIV_DESCR expressed during embryogenesis and provides key signal … and ligand of c-kit receptor CD117.
(Figure 2)
D1
D1:RysannDB
dbr:Cell_growth
. . .
dbr:Stem_cell_factor
. . .
dbr:CD117
D2
D2:TagME
cell growth DIV_NAME cell reproduction DIV_DESCR is used in the contexts of cell development …
stem cell factor DIV_NAME KL-1 DIV_DESCR is a cytokine that binds to the c-KIT receptor…
CD117 DIV_NAME C-Kit DIV_DESCR is a receptor tyrosine kinase protein that ..
D1U1
D1:RysannDB
U1:RysannMD
D1U2
D1:RysannDB
U2:NOBLECoder
D2U1
D2:TagME
U1:RysannMD
D2U2
D2:TagME
U2:NOBLECoder
SKOS:
exactMatch
closeMatch
dbr:CD117
dbr:Stem_cell_factor
RDFS:
seeAlso
dbr:Cell_growth
…

Figure 1: Pipeline for linking UMLS concepts to DBpedia using circular resolution method

METHODS

Algorithm

We pair four semantic annotation tools to perform link discovery between UMLS and DBpedia. Two pairings of annotators link UMLS concepts to DBpedia resources while the remaining pair links from DBpedia to UMLS concept-unique-identifiers (CUI). We label the DBpedia annotators and the UMLS annotators as D1 and D2, and U1 and U2, respectively.

Figure 1 outlines our link discovery method. The method starts with a UMLS concept of *Stem Cell Factor* (C0143630). The first step is to obtain the concept definition from UMLS ("*expressed during embryogenesis and provides key signal in multiple aspects of mast cell differentiation and function; hematopoietic growth factor and ligand of c-kit receptor CD117"*). Next, we construct a query string with all known labels and aliases for this UMLS concept and concatenate it with the concept definition, as shown in Table 1 *(left)*.

Table 1: Query string constructed for the UMLS concept *Stem Cell Factor* C0143630 *(left)* and DBpedia resource *Stem_cell_factor (right)*.

stem cell factor **DIV_NAME** mast cell growth factor **DIV_NAME** kit Ligand **DIV_NAME** steel factor **DIV_NAME** c-kit ligand **DIV_NAME** scf **DIV_DESCR** *expressed during embryogenesis and provides key signal … and ligand of c-kit receptor CD117.*	stem cell factor **DIV_NAME** steel factor **DIV_NAME** KITLG **DIV_NAME** KIT ligand **DIV_DESCR** *Stem cell factor (also known as SCF, KIT-ligand, KL, or steel factor) is a cytokine that binds to the c-KIT receptor (CD117).*

The query string is partitioned by a placeholder DIV_DESCR. This placeholder is used to divide the query string into two parts: labels with aliases (left-side) and UMLS definition (right-side). The right side is used by the semantic annotators to disambiguate the aliases on the left side of the placeholder. Similarly, the labels and aliases are kept separated from each other using a placeholder DIV_NAME to discourage semantic annotators from seeing incorrect multi-word n-grams by chance because of aliases situated next to each other. The generated query string is passed through two DBpedia semantic annotators (D1 and D2), each of which returns entity links to DBpedia resources (Step 3). The DBpedia resources found to the left of the DIV_DESCR placeholder are collected as *link candidates*. For each of these link candidates, a new query is constructed, also shown in Table 1, but using the labels, aliases and the abstract from DBpedia (Step 4). Each of these newly generated queries (from D1 and D2 link candidates) are then passed onto two UMLS semantic annotators (U1 and U2) in order to produce four UMLS annotated result sets: D1U1, D1U2, D2U1, and D2U2 (Step 5). Given these four result sets, we examine the UMLS annotations that appear to the left of the DIV_DESCR placeholder looking for an annotation with the CUI that we began with in Step 1, (i.e: C0143630). If such an annotation exists, then the candidate DBpedia resource is set aside to be later identified as either skos:exactMatch or skos:closeMatch (Step 6). Those candidates that do not produce the same CUI as the one used in Step 1 are delegated as having the weaker rdfs:seeAlso relationship.

In order to reduce disambiguation errors on the rdfs:seeAlso candidates, we discard those DBpedia resources that do not circularly resolve to *any* UMLS concepts in all four pairings of the annotators. In other words, all four pairings (D1U1, D1U2, D2U1, D2U2) must agree that the DBpedia resource resolves to some UMLS concept in order for the resource to remain as an rdfs:seeAlso relation.

Lastly, the skos:exactMatch/closeMatch set is separated into skos:exactMatch and skos:closeMatch relations by computing a *Jaccard coefficient* on all concept labels and aliases then testing for a minimum threshold. Formally, suppose UMLS concept CUI and DBpedia resource RES are related using *exact/close-match* as determined by our method (Figure 1). Let C and T be the set of all aliases/labels for CUI and RES, respectively. Let function A(s) return a set of individual characters from string s. Then, CUI is a skos:exactMatch to RES, if some label/alias of C and T meets the minimum threshold:

$$\max_{c\epsilon C,t\epsilon T}\frac{A(c)\cap A(t)}{A(c)\cup A(t)} \geq Threshold \qquad (1)$$

We will show later in the paper that our method is not sensitive to specific threshold values.

We name the above method *circular resolution* given the fact that we begin with a UMLS concept (C0143630); annotate a query string (composed of label+aliases+definition) with DBpedia resources; construct a similar query string for each of the returned DBpedia resources; then annotate these DBpedia query strings using UMLS semantic annotators hoping to loop back to the original UMLS concept (C0143630). We complete the method with Equation 1 to produce the results as shown in Figure 2.

rdfs:seeAlso —
dbpedia:Cell_growth
dbpedia:Coagulation
dbpedia:Chemical_reaction
dbpedia:Stem_cell
dbpedia:Ligand
dbpedia:Embryogenesis
dbpedia:Cell_(biology)
dbpedia:Gene_expression
dbpedia:Mast_cell

skos:closeMatch —
dbpedia:SCF_complex
dbpedia:CD117

skos:exactMatch —
dbpedia:Stem_cell_factor

Figure 2: The result of link discovery for *Stem Cell Factor* (C0143630) using the circular resolution method and Jaccard coefficient based [close|exact]-match classification.

Evaluation

To evaluate the effectiveness of our method, we queried Wikidata[8] for all entries that have a UMLS mapping to DBpedia. The query returned 5,006 entries. We disregarded mappings whose UMLS CUIs did not appear in our installation of UMLS because of licensing restrictions on the Metathesaurus. The final size of our testing set (ground truth) was 4,899 entries.

We performed an extensive search of the literature for reports on existing mappings of UMLS concepts to DBpedia that could serve as a benchmark for our algorithm and mapping. However, we found no such mapping, which suggests that our mapping is the first publicly available one. We also extensively searched for existing software tools that we could use to evaluate our algorithm and mapping. This proved quite difficult as we encountered numerous issues ranging from the lack of documentation to the unavailability of the systems themselves. Nonetheless, despite these issues, we were able to set up an additional baseline for comparison by using two annotators, i.e., RysannDB [24] and TagME [25], which have been used for biomedical named entity recognition. Henceforth, we evaluate our cooperative circular resolution algorithm by comparing it against RysannDB and TagME annotators, using the Wikidata ground truth.

RESULTS

We first focus our experiments on the output produced when only DBpedia annotators are used. In particular, we used RysannDB[9] [24] as D1, and TagME[10] [25] as D2. Figure 3 shows why annotating UMLS definitions with DBpedia annotators alone would be ineffective.

[8] https://www.wikidata.org
[9] http://denote.rnet.ryerson.ca/RysannDB
[10] https://tagme.d4science.org/tagme

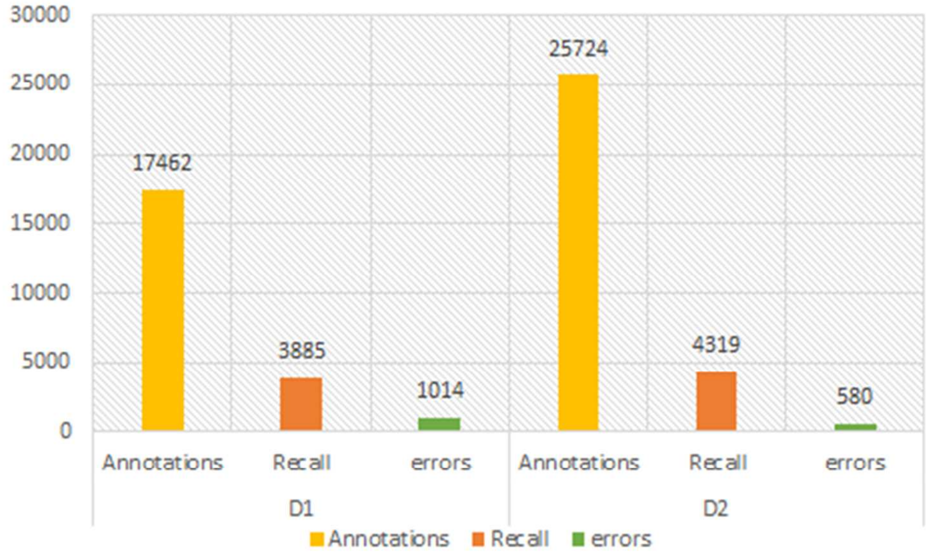

Figure 3: Counts of links produced by RysannDB (D1) and TagME (D2) when annotating the Wikidata ground truth. Includes counts of matching (Recall) and non-matching (errors) links.

RysannDB (D1) offered 17,462 entity links to DBpedia of which 3,885 matched the Wikidata ground truth. TagME (D2) produced 25,724 links with 4,319 matching links. Note that the Wikidata ground truth only contains 4,899 entries. The matching counts of 3,885 and 4,319 measure the recall, whereas precision is negatively affected by the additional 13,577 and 21,405 links provided by the two annotators.

Next, we pair D1/D2 with UMLS annotators RysannMD[11] [23] (U1) and Noble Coder[12] [26] (U2) to produce pairings of D1U1, D1U2, D2U1, and D2U2. Figure 4 shows how each pairing separately placed the ground truth into skos:exactMatch, skos:closeMatch, rdfs:seeAlso, or neither (disambiguation or recall error) using circular resolution.

From among the four pairings, the pairing of TagME and RysannMD (D2U1) was the most effective at linking UMLS to DBpedia with a 77.82% recall in identifying ground truth mappings as the expected skos:exactMatch relationship type. This pairing also achieved the smallest number of errors at 12.14%. The next best performing pair based on RysannDB and RysannMD (D1U1) achieved a recall of 71.30% with an error of 20.80%. Although the pairings of D1U2 and D2U2 performed weaker with a 52.62 and 58.77 percent recall agreement, it will be shown in Figure 5 that collectively they contribute to producing

[11] http://denote.rnet.ryerson.ca/RysannMD
[12] http://noble-tools.dbmi.pitt.edu

a better result. This is because each individual pairing provides some unique mappings that the others do not.

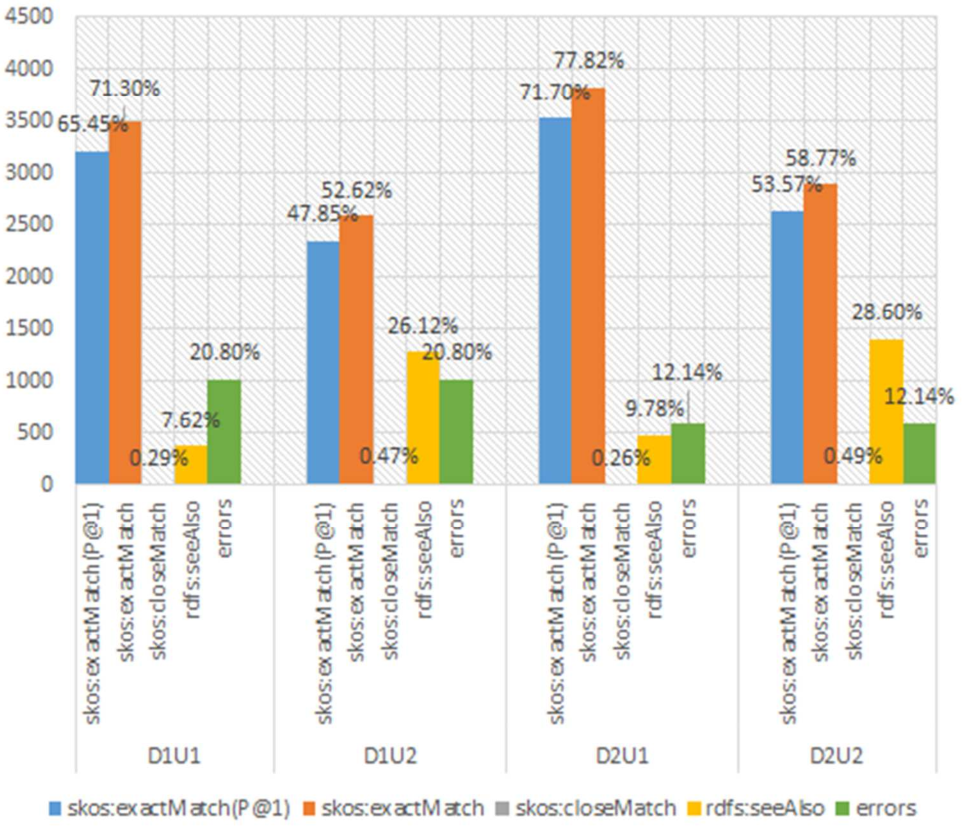

Figure 4: Count and percentage of ground truth mappings resolved as *skos:exactMatch*, *skos:closeMatch*, *rdfs:seeAlso*, or neither (error) against Wikidata including precision-at-1 for each annotator pairing.

We include in our analysis a precision-at-one (P@1) metric on the skos:exactMatch type to better judge the effectiveness of our circular resolution method. Specifically, the Wikidata ground truth assumes a 1-to-1 skos:exactMatch mapping between a UMLS Concept and a DBpedia resource. However, our technique may return multiple skos:exactMatch links for a single UMLS concept. Consequently, we report on the method's performance when a 1-to-1 mapping is strictly required by selecting the resource with the highest Jaccard coefficient that also meets the minimum threshold (Equation 1). We found our aforementioned top pairings of D1U1 and D2U1 still bested D1U2 and D2U2 with a precision-at-one of 65.45% and 71.70%.

Next, as per our pipeline (Figure 1), we pool together the mappings of each of the four pairings as a single solution, then report on skos:[exact|close]Match, rdfs:seeAlso, errors, and P@1 in Figure 5. Our findings show this combined mapping performs best in exactMatch (recall), errors, and exactMatch (P@1) than any of the individual pairings.

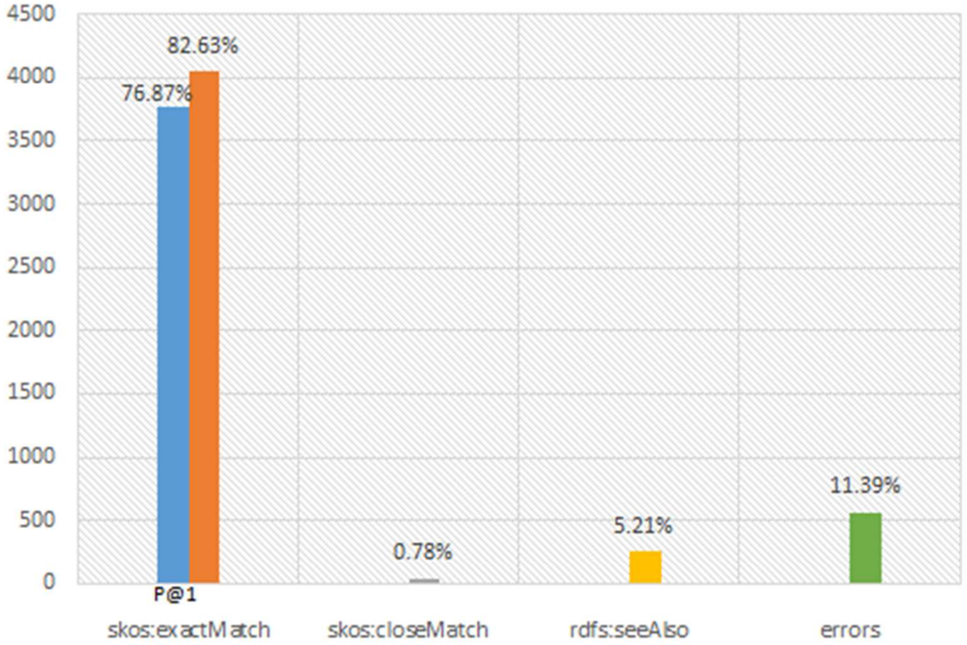

Figure 5: Count of ground truth mappings resolved as *skos:exactMatch*, *skos:closeMatch*, *rdfs:seeAlso*, or neither (error) including precision-at-1.

We conclude our tests by examining the sensitivity of our approach to the threshold for the Jaccard coefficient introduced in Equation 1. We show how the threshold affects precision and recall on skos:exactMatch classifications when a 1-to-1 UMLS concept to DBpedia resource mapping is required (i.e.: high precision P@1), and also when multiple DBpedia resources are allowed to link to a single UMLS concept, i.e., high recall. As shown in Figure 6, when the threshold value is set to zero, we observed 3,767 correctly mapped concepts at P@1 versus 4,086 correctly linked when a 1-to-many mapping is allowed. There was no change when the threshold was set to 0.25 and a negligible change of 1 exact-match to a close-match reclassification at a threshold of 0.5. Changes occurred when the threshold was set to 0.95 when a difference of 22 and 34 exact matches were observed. Furthermore, when the threshold was set to one, 27 and 39 exact match relationship changes were observed. The impact of varying the threshold from zero to one results in an overall performance change of around 0.5%; hence showing insensitivity to the threshold. From these results, we can conclude that the four annotators (D1/D2/U1/U2) are effectively leveraging their semantic capabilities to provide high quality candidates for close/exact-match determination and thus our method is relatively stable with respect to any chosen Jaccard threshold. Consequently, the best configuration would be utilizing Equation 1 to solely rank the candidates (using value zero as the threshold) then selecting the highest computed Jaccard for a 1-to-1 exact match (i.e. P@1).

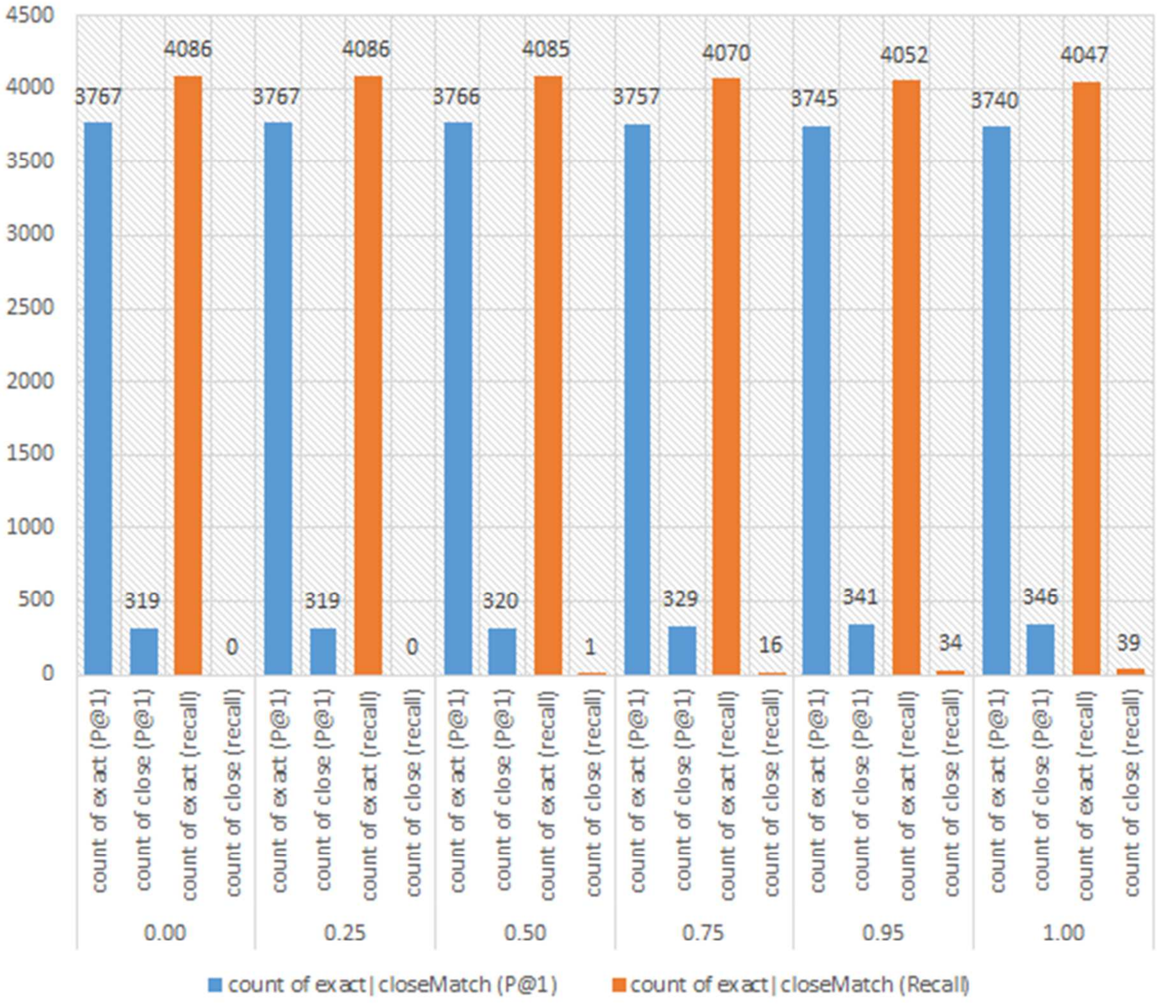

Figure 6: Counts on the number of exact/close matches with Jaccard threshold of 0 (no threshold), 0.25, 0.50, 0.75, 0.95 and 1.00.

DISCUSSION

Ground Truth Error Analysis

Our method achieved noticeable recall (83%) and precision scores (77%) during experimentation using the Wikidata ground truth benchmark. However, it did make mistakes depending on how an expected exact match concept was classified at various stages of the pipeline. We classify these errors as follows:

(1) *Candidate Selection Omission*. The DBpedia annotators (D1/D2) did not select the correct resource as a candidate. The outcome is that the correct resource does not appear as an exact match, close match, or see also.

(2) *Failed to Promote Error*. The UMLS annotators (U1/U2) did not produce any links from the candidate resource back to the target UMLS concept. In this case, the correct resource remains as an rdfs:seeAlso.

(3) *Failed to Meet Threshold*. Annotators U1/U2 correctly promoted a resource as a skos:closeMatch or skos:exactMatch but the correct resource either failed to meet the threshold in Equation 1 or a higher calculated Jaccard for the wrong concept was selected for P@1. This results in the correct resource being classified as skos:closeMatch.

(4) *Wrongly Promoted and Failed Jaccard Filtering*. A wrong concept was incorrectly promoted by U1/U2 and satisfied the Jaccard threshold or best P@1. This leads to linking the UMLS concept to an incorrect DBpedia resource as a skos:exactMatch (disambiguation error).

Table 2 summarizes the counts of the errors encountered during ground truth testing with ten examples for each error type. For example, CUI concept C0001815 "*Primary Myelofibrosis*" failed as a type (3) error resulting in a close match classification. This same CUI also suffered a type (4) error as it was wrongly linked to DBpedia resource *CIMF-FM*. The reader is encouraged to use our online browser[13] to further investigate each of these errors.

[13] http://denote.rnet.ryerson.ca/umlsMap/browser

Table 2: Summary of Circular Resolution error counts (type 1-4) with showcase examples of expected ground truth (G.T) and circular resolution answer (C.R).

(1) Candidate Selection Omission	(2) Failed to Promote Error	(3) Failed to meet Threshold	(4) Wrongly Promoted + Jaccard Filter
388 (No Link)	255 (rdfs:seeAlso)	38 (skos:closeMatch)	170 (error)
Sample CUIs			
C0496758	C2931205	C0031946	C0751782
C0302182	C0006111	C0153241	C0039753
C2937300	C0008684	C0041341	C0020433
C0153620	C0155937	C1274184	C0795690
C0022441	C1854540	C0019284	C0741160
C0023234	C2607929	C0018553	C0026697
C0025534	C1412004	C0266611	C0917990
C0477373	C1335473	C0001815	C0032290
C0795950	C1514284	C0007134	C0072826
C1841679	C0279607	C0343065	C1337224
Showcase Example			
CUI: C0477373 “Other forms of migraine” G.T: Familial_hemiplegic_migraine C.R.: no entity link	CUI: C1514284 “Potassium Deficiency Disorder” G.T: Hypokalemia C.R: Linked as rdfs:seeAlso	CUI: C0001815 “Primary Myelofibrosis” G.T: Myelofibrosis C.R: CIMF-FM (exact) Myelofibrosis (close)	

Mapping the UMLS

We applied our method to the UMLS Metathesaurus to produce 221,690 skos:exactMatch, 26,276 skos:closeMatch, and 6,784,322 rdfs:seeAlso relations.The total number of concepts in our license-free version of the UMLS was 2,397,167. This gives a percentage of mapping from the UMLS to DBpedia for skos:[close|exact]Match of 10.34 percent and an average of 2.83 rdf:seeAlso relationships per concept. Although this may seem a low percentage, consider that our ground truth from all of Wikidata contained only 5,006 mapped UMLS concepts compared to our 221,690 mappings (a factor of 50x increase). The difficulty in mapping a large portion of the UMLS as an exact match occurs largely because many concepts are so specific as to not have a corresponding entry in DBpedia, as illustrated in Table 3. This is not very surprising to those familiar with UMLS. In order to gain further insight, we performed a simple experiment in which we surmised that the one-word concepts in UMLS were more likely to have a corresponding exact match DBpedia entry than those comprising two or more words. To further challenge our method, we excluded those one-word concepts that appeared directly within the DBpedia URL itself thus making it more difficult for the annotators to perform the alignment (e.g., *C0018081:Gonorrhea* mapped to *dbpedia: Gonorrhea* was excluded from this experiment). A cursory inspection of a random sampling of the 24,179 one-word mappings revealed good results with success and error rates equivalent to those observed in Figure 5 and Table 2. For example. our method correctly mapped *C0001429:Adenolymphoma* with *dbpedia:Warthin's_tumor,* but mistakenly matched *C1174791:Basen* to *dbpedia:Basen,_Armenia*. We have provided this one-word mapping as a supplementary document for further inspection.

Table 3: Two examples of *rdfs:seeAlso* mappings where no exact match is available.

C3175196 "*Other people frequently tell me that what I've said is impolite even though I think it is polite: d:Pt: ^Patient: Ord: PhenX*" **rdfs:seeAlso —** dbpedia:Taboo dbpedia:Time dbpedia:Patient dbpedia:Thought dbpedia:Level_of_measurement	C0370538 "*Punch graft for hair transplant; more than 15 punch grafts*" **rdfs:seeAlso —** dbpedia:Bone_grafting dbpedia:Hair dbpedia:Organ_transplantation dbpedia:Hair_transplantation dbpedia:Graft_(surgery)

Maintaining the UMLS Mapping

From the perspective of the choice of the semantic annotators, RysannDB (D1) and TagME (D2) were selected as the DBpedia linkers because of their accuracy and speed of processing natural language text. Speed is a particular concern since our goal was to map the entire UMLS to DBpedia. Some other well-known annotators, although of comparable accuracy, are too slow to be practical for this task. The same consideration was given to the choice of RysannMD (U1) and Noble Coder (U2) based on the findings in [24].

The time to map UMLS to DBpedia required approximately 60 hours of processing for each pairing (D1U1, D1U2, D2U1, D2U2) on an Intel 3.00GHz Xeon CPU based server with 128GB of RAM. Although this may seem time intensive, one should consider the following:

(1) Our implementation of circular resolution was focused on link discovery challenges, not on processing time optimization. Efficiency-oriented implementations would execute the processing of pairs D1/D2 and U1/U2 concurrently thus reducing the mapping time by a factor of four. Further improvements can be gained by dividing the UMLS database into smaller datastores and processing in parallel.

(2) Updates of the mapping require the processing of only new UMLS entries allowing for incremental updates.

(3) Like other open datasets, the burden of (1) and (2) falls to the authors of this work as the dataset maintainers. We intend to maintain this dataset and make it available through our website and officially through the LOD cloud.

Alternative Approaches

Link discovery and instance matching is an active area of research, with many open challenges. A comprehensive survey by Nentwig et al. gives a good summary of the current state-of-the-art [23]. In this survey, nine out of eleven examined frameworks could only determine owl:sameAs relationships. The remaining frameworks (Silk [15] and LIMES [16]), do support additional link types through heuristic rules. However, the user is responsible for manually constructing the necessary heuristic patterns for detecting a particular relationship type, e.g, rdfs:seeAlso. In contrast, our method operates at a higher level of abstraction relying on underlying semantic annotation engines. This allows our method to easily take advantage of a wide combination of techniques that have already been incorporated into existing semantic annotators by choosing different annotators to fill in the role of D1, D2, U1 and U2. Furthermore, the heuristic rules approach taken by Silk and LIMES may not be interchangeable between different pairs of KBs. That is, rules designed to map from KB1 to KB2 may not be the same rules needed to map from KB1 to KB3 even for the same link type. Comparatively, our method performs the alignment by only considering textual information from readily available concept labels/definitions and through the use of the natural language processing capabilities of existing semantic annotators. It should be noted, however, that the heuristic rules approach undertaken by Silk and LIMES does allow for flexibility in the relationship type sought after whereas our method is limited to skos:[exact|close]Match and rfds:seeAlso;- an important point in the conceptual distinction between Silk and LIMES and our work. Both Silk and LIMES are customizable and extensible frameworks on top of which specific link discovery processes are implemented to interconnect different datasets. Both of these frameworks are primarily developed to allow experts to design mapping pipelines from existing components that are shipped with the two frameworks or can be added to the frameworks as third party add-ons. However, our work focuses on one specific mapping process and hence would not be considered as an extensible framework. In this light, circular resolution could be integrated into the LIMES or Silk pipeline that could prove valuable for a wider range of mapping tasks.

Lastly, we considered numerous designs for circular resolution before settling on the method proposed here. One such consideration involved the treatment of the primary label and alternative names of a concept as separate annotation problems, which would then be merged. This approach would have eliminated the use of the separation tokens, i.e., DIV_NAME and DIV_DESCR. Details of this alternative method, and the reason for its dismissal, are given in a supplementary document (Appendix A).

CONCLUDING REMARKS

In this paper, we have presented a method, called circular resolution, to map UMLS concepts to DBpedia resources using rdfs:seeAlso, skos:closeMatch, and skos:exactMatch relations. Our technique reports a recall of 83% with 77% precision-at-one when benchmarked against Wikidata. A full UMLS to DBpedia mapping is also made publicly available. In addition, we provide an online browser to easily explore the mappings and a RESTful interface for querying the mappings[14]. We hope that this mapping can become

[14] http://denote.rnet.ryerson.ca/umlsMap

an integral part of the *Linked Open Data cloud* and facilitate the effective interchange and integration of different knowledge bases with medical and biomedical knowledge bases. To this end, our future work includes creating UMLS mappings for the various ontologies openly available through "*The Open Biological and Biomedical Ontology (OBO) Foundry*"[15] which provides open access to medical and biological vocabularies.

[15] http://www.obofoundry.org/

Funding Statement

This work was supported by the *Natural Sciences and Engineering Research Council of Canada* under grant number RGPIN-2015-06118

Competing Interests Statement

The authors of this work have no competing interests to declare.

Contributorship Statement

The authors declare that this manuscript is a product of original work and each author contributed to the design and interpretation of the results. Furthermore, the authors have critically evaluated the content before final approval for publication. The authors are accountable for all aspects of this work and believe in the accuracy of their results, interpretation thereof, and content of this manuscript.

REFERENCES

[1] Lehmann J, Isele R, Jakob M, et al. DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. Semantic Web Journal 2012;6(2):167-195.

[2] Heath T, Bizer C. Linked Data. Evolving the Web into a Global Data Space (1st edition). Synthesis Lectures on the Semantic Web: Theory and Technology. San Rafael, California, USA: Morgan & Claypool 2011: 1-136.

[3] Pai VM, Rodgers M, Conroy R, et al. Workshop on using natural language processing applications for enhancing clinical decision making: an executive summary. J Am Med Inform Assoc 2014;21(e1): e2-e5.

[4] Friedlin J, McDonald, CJ. An evaluation of medical knowledge contained in Wikipedia and its use in the LOINC database. J Am Med Inform Assoc 2010;17(3): 283-287.

[5] Dessì, N., & Atzori, M. (2017). Is Wikipedia a Latent Gene Ontology? In 2017 IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE) (pp. 164–169). https://doi.org/10.1109/WETICE.2017.19

[6] Yamamoto, Y., Yamaguchi, A., & Yonezawa, A. Building Linked Open Data towards integration of biomedical scientific literature with DBpedia. *Journal of Biomedical Semantics*, 2013;4:8. https://doi.org/10.1186/2041-1480-4-8

[7] Laurent MR, Vickers TJ. Seeking Health Information Online: Does Wikipedia Matter?. J Am Med Inform Assoc 2009;16(4):471-479.

[8] Wiesner M, Pfeifer D. Health Recommender Systems: Concepts, Requirements, Technical Basics and Challenges. International Journal of Environmental Research and Public Health 2014;11(3):2580-2607.

[9] Park, M., Sampathkumar, H., Luo, B., & Chen, X. w. (2013). Content-based assessment of the credibility of online healthcare information. In 2013 IEEE International Conference on Big Data (pp. 51–58). https://doi.org/10.1109/BigData.2013.6691758

[10] Keselman A, Smith CA, Divita G, et al. Consumer Health Concepts That Do Not Map to the UMLS: Where Do They Fit?. J Am Med Inform Assoc 2008;15(4):496-505.

[11] Roberts K & Demner-Fushman, D. Interactive use of online health resources: a comparison of consumer and professional questions. J Am Med Inform Assoc 2016;23(4):802-811.

[12] Mrabet, Y., Kilicoglu, H., Roberts, K., & Demner-Fushman, D. Combining Open-domain and Biomedical Knowledge for Topic Recognition in Consumer Health Questions. *AMIA Annual Symposium Proceedings*, 2016; 914–923. TODO

[13] Ernst, P., Siu, A., Milchevski, D., Hoffart, J., & Weikum, G. (2016). DeepLife: An Entity-aware Search, Analytics and Exploration Platform for Health and Life Sciences. In S. Pradhan, & M. Apidianaki (Eds.), Proceedings of ACL-2016 System Demonstrations (pp. 19-24). Stoudsbourg, PA: ACL. doi:10.18653/v1/P16-4004

[14] Ernst, P., Siu, A., Weikum, G. 2015. Knowlife: a versatile approach for constructing a large knowledge graph for biomedical sciences. BMC Bioinformatics 16(1):1–13.

[15] Volz J., Bizer C., Gaedke M., Kobilarov G. Silk – A Link Discovery Framework for the Web of Data. Workshop on Linked Data on the Web (LDOW2009), 2009.

[16] Ngomo A., Auer S.. LIMES: a time-efficient approach for large-scale link discovery on the web of data. In Proceedings of the Twenty-Second international joint conference on Artificial Intelligence (IJCAI'11), 2011;2312-2317. DOI=http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-385

[17] Tilahun, B., Kauppinen, T., Keßler, C. and Fritz, F. Design and development of a linked open data-based health information representation and visualization system: potentials and preliminary evaluation. *JMIR medical informatics*, *2*(2), 2014.

[18] Bing H., Tang J., Ding Y., Wang H., Sun Y., Shin J., Chen B., Moorthy G.,Qiu J., Desai P., Wild D. Mining Relational Paths in Integrated Biomedical Data. PloS one 2011.

[19] Luciano J., Andersson B., Batchelor C., Bodenreider O., Clark T., Domarew C., Gambet T., Harland L, et. al. The Translational Medicine Ontology and Knowledge Base:Driving personalized medicine by bridging the gap between bench and bedside. J Biomed Semantics. 2011; doi: 10.1186/2041-1480-2-S2-S1.

[20] Saleem M., Padmanabhuni S., Ngomo A., Almeida J., Decker S., Deus H. Linked Cancer Genome Atlas Database. Proceedings of the 9th International Conference on Semantic Systems. I-SEMANTICS '13. 2013;129-134.

[21] Lee, L.H., Groß, A., Hartung, M., Liou, D.M. and Rahm, E. A multi-part matching strategy for mapping LOINC with laboratory terminologies. *Journal of the American Medical Informatics Association*, *21*(5), pp.792-800, 2014.

[22x] Kahn, C.E. Integrating ontologies of rare diseases and radiological diagnosis. *Journal of the American Medical Informatics Association*, *22*(6), pp.1164-1168, 2015.

[23] Nentwig M, Hartung M, Ngonga Ngomo AC, Rahm E. A survey of current Link Discovery frameworks. Semantic Web 2017;8(3):419-436.

[24] Cuzzola, J., Jovanovic, J., Bagheri, E., RysannMD: A Biomedical Semantic Annotator Balancing Speed and Accuracy, Journal of Biomedical Informatics. 2017;doi: http://dx.doi.org/10.1016/j.jbi.2017.05.016

[25] Ferragina P., Scaiella U. TAGME: on-the-fly annotation of short text fragments (by wikipedia entities). In Proceedings of the 19th ACM international conference on Information and knowledge management (CIKM '10). ACM, New York, NY, USA, 2010;1625-1628. doi:http://dx.doi.org/10.1145/1871437.1871689

[26] Tseytlin E, Mitchell K, Legowski E, Corrigan J, Chavan G, Jacobson RS. NOBLE - Flexible concept recognition for large-scale biomedical natural language processing. BMC Bioinformatics. 2016;17:32. doi: 10.1186/s12859-015-0871-y

APPENDIX A: AN ALTERNATIVE IMPLEMENTATION OF CIRCULAR RESOLUTION

During conceptualization of our method, we faced the problem of alternative names (aliases) for UMLS and DBpedia concepts. For example, the UMLS concept C0143630 has six names: stem cell factor, mast cell growth factor, kit ligand, steel factor, c-kit ligand, and scf. DBpedia offers four names: stem cell factor, steel factor, KITLG, and KIT ligand. Our initial thought was to annotate each of these names separately using the same UMLS/DBpedia description, and subsequently combine the individual results. Although initially this seemed logical, we quickly came to realize that this strategy was flawed in three key areas.

(1) Speed and (2) Complexity. Using UMLS concept C0143630 as our working example, we define the set N as the list of possible concept names N=[stem cell factor, mast cell growth factor, kit ligand, steel factor, c-kit ligand, scf.].Furthermore, we define E as the UMLS explanation for this concept: E="*expressed during embryogenesis and provides key signal in multiple aspects of mast cell differentiation and function; hematopoietic growth factor and ligand of c-kit receptor CD117*". Then, we need to perform six individual annotations of E|N1, E|N2, E|N3, E|N4, E|N5, E|N6, in comparison to a single annotation of E|N when using DIV_NAME and DIV_DESCR separator tokens. This significant increase in processing time and computational resources intensifies as the circular resolution process proceeds through the pipeline, as illustrated in Figure A-1.

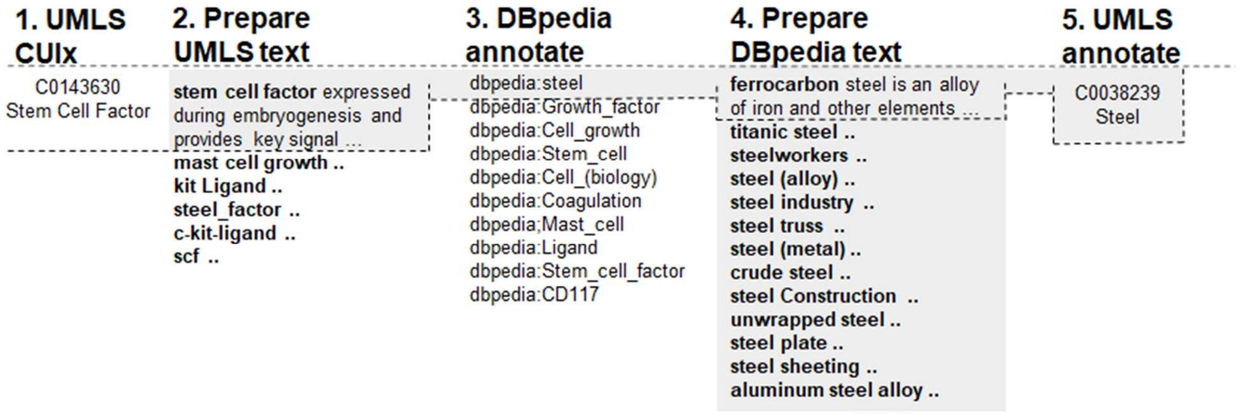

Figure A-1: Revised pipeline of the circular resolution method without DIV_NAME/DIV_DESCR boundaries.

At stages 2 and 3, the annotation of the text of E|N_1="stem cell factor" results in ten candidates, from dbpedia:steel to dbpedia:CD117, each requiring its own processing with a DBpedia annotator (D1). The DIV_NAME/DIV_DESCR method would have only required one query of text for D1. Next, consider the first of these ten candidates (dbpedia:steel) which has thirteen alternative names, from "ferrocarbon" to "aluminum steel alloy". This, too, would require separate textual descriptions, each individually annotated at stages 3 and 4 using a UMLS annotator (U1). In contrast, the DIV approach would have required a single query text for U1. This process repeats for all candidates of stages 2 and 5 and is multiplied by a factor of three (3x) for the other combinations of UMLS and DBpedia annotators: U1D2, U2D1, and U2D2. It is clear that this strategy is significantly slower than our DIV_NAME/DIV_DESCR consolidated approach. Furthermore, even if this was not computationally prohibitive, we suspect it would produce worse results than our current design for reasons described next.

(3) Reduced Accuracy. The alternative method suffers from a well known problem of the independence assumption. Consider the concept *Rhinovirus (C0035473),* which has alias name *cold*. Using the previous notation of N=[rhinovirus, cold] and E="*A genus of single-stranded positive sense RNA viruses containing a single RNA molecule*", the alternative implementation would need to calculate individual probabilities of P(E|rhinovirus) and P(E|cold), then multiply them together to obtain the final probability score. This calculation, known as Naive Bayes, assumes independence of terms "rhinovirus" and "cold", which is clearly not the case. The annotators (D1,D2,U1,U2) are more likely to associate the term "cold" with "illness" instead of the concept "low temperature" in the context of the co-occurring term "rhinovirus".

To sum up, by calculating P(E|rhinovirus) and P(E|cold) separately, we force a simple Naïve Bayes determination of ambiguity, thus preventing the annotators from using more advanced algorithms that consider the co-occurrence of terms when computing the final probability. It is this reduction of the problem to a Naïve Bayes solution that makes the one-label-per-document implementation problematic and ultimately rejected.