From General to Specialized Domain: Analyzing Three Crucial Problems of Biomedical Entity Disambiguation

Stefan Zwicklbauer, Christin Seifert, Michael Granitzer

University of Passau, Passau 94032, Germany,
forename.surname@uni-passau.de

Abstract. Entity disambiguation is the task of mapping ambiguous terms in natural-language text to its entities in a knowledge base. Most disambiguation systems focus on general purpose knowledge bases like DBpedia but leave out the question how those results generalize to more specialized domains. This is very important in the context of Linked Open Data, which forms an enormous resource for disambiguation. We implement a ranking-based (Learning To Rank) disambiguation system and provide a systematic evaluation of biomedical entity disambiguation with respect to three crucial and well-known properties of specialized disambiguation systems. These are (i) entity context, i.e. the way entities are described, (ii) user data, i.e. quantity and quality of externally disambiguated entities, and (iii) quantity and heterogeneity of entities to disambiguate, i.e. the number and size of different domains in a knowledge base. Our results show that (i) the choice of entity context that is used to attain the best disambiguation results strongly depends on the amount of available user data, (ii) disambiguation results with large-scale and heterogeneous knowledge bases strongly depend on the entity context, (iii) disambiguation results are robust against a moderate amount of noise in user data and (iv) some results can be significantly improved with a federated disambiguation approach that uses different entity contexts. Our results indicate that disambiguation systems must be carefully adapted when expanding their knowledge bases with special domain entities.

Keywords: Entity Disambiguation, Learning to Rank, Linked Data, Semantic Web

1 Introduction

Semantically structured information like Linked Data exhibits huge potential for improving unstructured information management processes in different domains like the Web, enterprises or research. In particular, textual information can be linked to concepts found in the Linked Open Data (LOD) cloud to improve retrieval, storage and analysis of large document repositories. Entity disambiguation algorithms establish such links by identifying the correct semantic meaning from a set of candidate meanings, referred to as the knowledge base (KB), to a selected text fragment, also called surface form. For instance, given a sentence with surface form “Ford”, an entity disambiguation algorithm determines whether the surface form refers to the actor (Harrison Ford), the 38th President of the United States (Gerald Ford), the organization (Ford Motor Company) or the place (Ford Island) [24].

Entity disambiguation has been studied extensively in the past 10 years. Most prior work focus on disambiguating entities of general KBs like Wikipedia and other encyclopedias [9, 12, 15, 18, 20]. Recent work takes on LOD data sets as KB, but still focuses on generic entities like cities, persons etc. [20, 18]. However, its results do not hold true for disambiguating entities of more specialized domains. When taking specialized entities from the LOD cloud, disambiguation is more difficult due to special domain characteristics. For instance, LOD data sets that contain biomedical entities often lack appropriate entity descriptions (e.g. genes) or provide domain-specifity (e.g. UniProt focuses on genes only). Overall, a systematic evaluation of specialized entity disambiguation w.r.t special domain properties with entities of the LOD cloud is missing.

In our work we first identify the following three crucial special domain properties:

1. entity context, i.e. the way how entities are described
2. user data, i.e. quantity and quality of externally disambiguated entities
3. quantity and heterogeneity of entities to disambiguate, i.e. the number and size of different domains in a KB

Further, to evaluate these special domain properties, we focus on the biomedical domain which is extensively represented by several large data sets in the LOD cloud. Biomedical entity disambiguation is a challenging task due to a considerable extent of ambiguity and thus has attained much attention in research in the last decade [24]. While many biomedical disambiguation algorithms apply common String matching approaches, we combine well-established disambiguation features in a ranking approach (Learning to Rank) to perform an in-depth evaluation of our special domain properties.

Overall, our **contributions** are the following:

- We provide a systematic evaluation of biomedical entity disambiguation with respect to entity context, user data as well as quantity and heterogeneity of entities.
- We show that the choice of entity context that is used to attain the best disambiguation results strongly depends on the amount of available user data.
- We show that entity contexts strongly affect disambiguation results with large-scale and heterogeneous KBs.
- We show that results are robust against a moderate amount of noise in user data.
- We show that by using a federated approach with different entity contexts some results can be improved significantly (Mean Reciprocal Rank as well as robustness against large-scale and heterogeneous KBs).

The remainder of the paper is structured as follows: In section 2 we identify and model the evaluated special domain properties. Section 3 describes the implementation of our disambiguation system. Section 4 analyzes the biomedical data set CALBC which is used in our evaluation. Section 5 presents experiments in form of an in-depth evaluation. In section 6 we review related work. Finally, we conclude our paper in section 7.

2 Problem Statement and Modeling

First, we identify the properties entity context, user data and quantity and heterogeneity of entities which resemble core properties for specialized disambiguation systems. Second, we introduce how we model these properties in the context of our work.

2.1 Identifying Important Properties of a Specialized Disambiguation System

Problem 1: Disambiguating domain-specific entities demands a specialized disambiguation system that covers the entire range of entities belonging to the respective domain. The creation of such a system includes the choice of a data set that describes all entities as effectively as possible. Generally, an entity can be defined intensionally, i.e. through a description, or extensionally, i.e. through instances and usage [13]. Intensional definitions can be understood as a thesaurus or logical representation of an entity, as it is provided by LOD repositories. Extensional definitions resemble information on the usage context of an entity, as it is provided by entity-annotated documents. Many disambiguation systems apply LOD repositories on general knowledge (e.g. DBpedia) due to its rich feature set (e.g. descriptions, relations). LOD repositories comprising special-domain entities regularly lack such features [24]. For instance, entities like "FV3-049L" in the UniProt KB lack extensive disambiguation-relevant descriptions or relations.

This raises the question of how disambiguation with intensional and extensional entity descriptions performs in specialized domain. Additionally, the question remains to which extent federated disambiguation with both entity contexts improves the results. We refer to both extensional and intensional entity descriptions, as **entity context**.

Problem 2: Extensionally constructed KBs contain information about the entities usage context in terms of entity-annotated documents. These textual documents contain words or phrases that were linked to their entities either manually by users or automatically by disambiguation systems. In specialized domains the quantity and quality of available annotated documents is generally very limited.

The question remains to which extent quantity and quality of annotated documents influence disambiguation with different entity contexts on specialized domains. We denote words or phrases and their mapping to entity identifiers as **user data**.

Problem 3: Several general-domain disambiguation systems use DBpedia as KB due to its wide-ranging and high quality entities. DBpedia also comprises a broad range of popular entities from several specialized domains (e.g. Influenza) but lacks very specific entities [19] (e.g. IIV3-011L gene). However, the LOD cloud offers several data sets comprising entities belonging to a specific subdomain. For instance, the UniProt KB contains genes/proteins only and therefore also contains very unpopular and rare occurring entities. To cover all entities of a specialized domain, we collect the entities of several LOD data sets. This may lead to (extremely) large and heterogeneous KBs.

The question remains how quantity and heterogeneity of entities affect disambiguation accuracy in specialized domains. In the following we refer to this property as the **quantity and heterogeneity of entities**.

2.2 Modeling the Properties in Context of a Biomedical Disambiguation System

After identifying important properties of a specialized disambiguation system, we focus on the biomedical domain which is perfectly suitable for our analysis. In the following we specify and model the properties (i) entity context, (ii) user data and (iii) quantity and heterogeneity of entities in context of our work.

Modeling Entity Context

Entities are described either extensionally or intensionally. We model these entity context forms as an *entity-centric* (intensional entity representation) or *document-centric* KB (extensional entity representation) which comprise disambiguation-relevant entity information extracted by the original data sets. Figure 1 illustrates our model. The edge between extensional data and entity-centric KB depicts the usage of user data in the entity-centric KB (e.g. surface forms, synonyms).

Formally, we define an *entity-centric* KB as

$$KB_{\text{ent}} = \{e_0, ..., e_n | e_i \in E, n \in \mathbb{N}\} \tag{1}$$

The set of all entities available in KB_{ent} is denoted as E, with e_i being a single entity. All entities $e_i \in KB_{\text{ent}}$ have a unique primary key ID which combines the name of the knowledge source as well as its identifier in the knowledge source. Additionally, a variable number of fields k contain domain-independent attributes, e.g., description, and domain-dependent information, e.g., the sequence length of genes. Formally we denote such an entity as $e_{\text{i}} = (ID, Field_1, ..., Field_k)$.

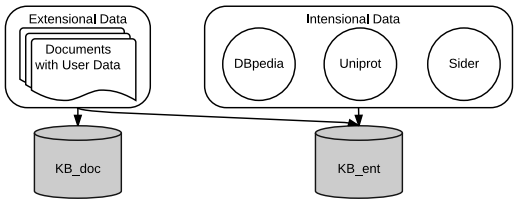

Fig. 1: Modeling entity-centric and document-centric KBs.

A *document-centric KB* is defined as

$$KB_{\text{doc}} = \{d_0, ..., d_n | d_i \in D, n \in \mathbb{N}\} \tag{2}$$

An entry d_i in a document-centric KB consists of the title, the content, both representing a text string, and a set of annotations $\{(t_i, \Omega_i)\}$. An annotation contains a surface form t and a set Ω with entity identifiers. These entity identifiers are referred by the respective surface form t. In the following, we denote an entry in a document-centric KB as $d_{\text{i}} = (Title, Content, \{(t_1, \Omega_1) \dots (t_k, \Omega_k)\})$.

Modeling User Data

In our work the set of all user annotations in natural-language documents is called user data. A user annotation consists of a textual representation t, the surface form, and an entity set Ω, which is referred by surface form t. Example 3 shows an annotation of surface form "H1N1", with the id denoting an entity's LOD resource:

...WHO declared <e id="UMLS:C1615607:T005:diso>H1N1</e> influenza... (3)

As depicted in Figure 1 user data is stored in both, entity-centric and document-centric KBs. In our work we assume that user data is readily available and provided by the underlying data set (cf. Section 4).

Modeling Large-scale and Heterogeneous KBs
Basically, increasing the heterogeneity within a KB is caused by adding entities from other domains. Hence, we distinguish between an *intra-specific* domain extension and an *inter-specific* domain extension. An intra-specific domain extension describes a KB enrichment with entities or documents from the same domain. In our case we add entities and documents from the biomedical domain (e.g. adding a gene database). In contrast a KB enrichment with documents or entities from other domains (e.g. DBpedia) describes an inter-specific domain extension.

3 Approach

To study the three properties of specialized domain disambiguation systems, namely the entity-context, user data, and the quantity and heterogeneity of entities to disambiguate, we create a disambiguation system. Figure 2 shows an overview of our system containing an entity-centric and document-centric disambiguation algorithm, both relying on their respective KB. The results of both approaches, which are ranked by means of Learning to Rank (LTR), are combined in a federated disambiguation approach.

In the following section we first describe the methods for disambiguation with an underlying entity-centric and document-centric KB. Second, we describe the LTR feature set in our algorithms. Finally, we describe our federated disambiguation approach.

3.1 Entity-Centric and Document-Centric Disambiguation

Our entity-centric and document-centric disambiguation algorithms can be described as ranking-based approaches for disambiguating entities e_i. Given a knowledge base KB that contains all available entity candidates, a surface form t as well as its context words c_t^λ (λ denotes the number of words in front of and after surface form t), we return a ranked list R of entities in descending score order, i.e.

$$R = rank(KB, t, c_t^\lambda) \tag{4}$$

Our **entity-centric disambiguation** approach uses a linear combination of a weighted feature set F_{ent} to compute a score $S_{e_i}^{ent}$ for each entity:

$$S_{e_i}^{ent} = w^\intercal f(e_i, t, c_t^\lambda) \tag{5}$$

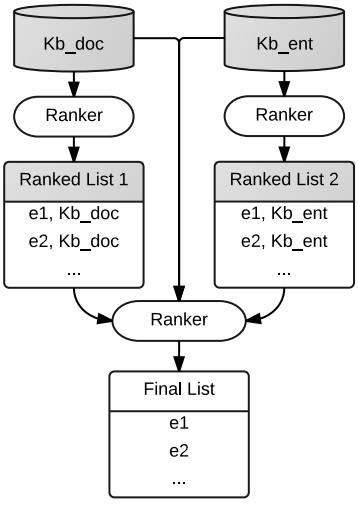

Fig. 2: Disambiguation System

Variable w denotes the weight vector for our feature set and function $f(e_i, t, c_t^\lambda)$ returns a vector containing the feature values of entity e_i with reference to surface form t and its context c_t^λ. The disambiguation result R consists of the Top-N scored entities.

Our **document-centric disambiguation** algorithm is similar to a K-Nearest-Neighbor classification using majority voting. First, we obtain a predefined number τ of relevant documents using the ranking function as defined in Equation 5 with another feature set. A relevant document should contain similar content as given by surface form t and surrounding context c_t^{λ}. The second step encompasses the classification step. We compute the score $S_{e_i}^{doc}$ for all referenced entities K in our queried document set T_τ:

$$S_{e_i}^{doc} = \sum_{j}^{T_\tau} p(e_i|d_j) \tag{6}$$

Probability $p(e_i|d_j)$ denotes the probability of entity e_i occurring in document d_j (with reference to all documents in KB_{doc}). To determine the probabilities of entities occurring in documents we apply a modified Partially Labeled Latent Dirichlet Allocation approach (PLDA) [14], which is similar to the approach of mining evidence for entity disambiguation [10]. Due to space constraints we refer the reader to the referenced papers for details. Again, the result list R consists of the Top-N scored entities. The quality of the results strongly depends on the number of annotated entities in the document set. Generally, when using a document-centric KB, user data must be available.

3.2 Feature Choice

In the following we describe our LTR feature set used for entity-centric and document-centric disambiguation. We distinguish between three feature sets: string similarity features, prior features and evidence features (cf. Table 1). Our document-centric algorithm uses string similarity features only (according to the data in the KB) while the entity-centric approach applies all.

Table 1: Overview of LTR features

Nr.	Feature
1	Jaro-Winkler distance between surface form and entity names
2	TF-IDF weight of surface form w.r.t all entity names
3	TF-IDF weight of surface form w.r.t all entity descriptions
4	TF-IDF weight of context w.r.t all entity names
5	TF-IDF weight of context w.r.t all entity descriptions
6	BM-25 weight of surface form w.r.t all entity descriptions
7	BM-25 weight of context w.r.t all entity descriptions
8	Prior: Occurrences of an entity
9	Sense Prior: Entity occurrences with a specific surface form
10	Co-occurrences: Entity-entity alignment
11	Term evidences: Entity-term alignment

String Similarity Features:
String similarity features are used in both disambiguation approaches. In the entity-centric approach we restrict our result list to those entities whose names or synonyms do not match with the surface form. For this purpose we choose the Jaro-Winkler distance [6] which is designed and best suited for short strings such as person names. Other features compute the similarity between the surface form and the entity names/synonyms as well as the entity description. Additionally, we determine the similarity between the context words and the entity names/synonyms as well as the entity description. We apply the Vector Space Model with TF-IDF weights and the Okapi BM25 model (cf. Table 1 features 2-7) for similarity computation. This similarity feature set attains the best results in our evaluation, but our approach leaves the option of choosing other metrics open.

In the document-centric approach we use the Vector Space Model (TF-IDF) and Okapi BM25 model to search for documents with similar content as given by the surface form and context words (feature 3, 5-7). TF-IDF and BM-25 weights of surface forms and surrounding context are computed w.r.t to the documents title and content. We omit the Jaro-Winkler distance as filter due querying documents instead of relevant entities. An in-depth explanation of these models is provided by [11].

Prior Features:
Generally, some entities (i.e. Influenza) occur more frequent than others (i.e. IIV3-011L gene) in documents. Thus, these popular entities provide a higher probability to reoccur in other documents. In our work the *Prior* $p(e_i)$ describes the a-priori probability that an entity occurs and was initially proposed by Philip Resnik [16]. A logarithm is used for this feature to damp high values. The *Sense Prior* $p(e_i|t)$ estimates the probability of seeing an entity with a given surface form [12]. All probabilities are computed by analyzing available user data.

Evidence Features:
The *Co-occurrence* feature Co_{e_i} considers context words of surface form t as potential surface forms. Basically, we assume that surface form t's real referent entity provides a higher probability to co-occur with potential but not yet disambiguated entities located in the surrounding context. First, we assume the context words c_t^λ of our surface form t to be surface forms of other entities. Hence, we compare the context words c_t^λ with all existing surface forms provided by available user data. If a context word c_j matches with one of these surface forms, we use this surface form's referent entity e_k and compute the probability of our entity candidate e_i occurring with e_k. For instance, the context word "influenza" of surface form t has already been used as surface form to address the entity "H1N1" in a document. Thus, "H1N1" constitutes a potential entity for our context word and we compute the probability of our entity e_i co-occurring with "H1N1":

$$Co_{e_i} = \sum_{c_j \in c_t^\lambda} \log(1 + \underset{e_k \in f(c_j)}{argmax}\, p(e_k|e_i)p(e_k|c_j)) \tag{7}$$

We investigate all context words c_t^λ to compute the feature score. Function $f(c_j)$ delivers a set of entities that have been annotated in combination with the possible "surface form" c_j in other documents. Further, $p(e_k|e_i)$ describes the probability of entity e_k co-occurring with our entity candidate e_i. Additionally, we take the sense prior $p(e_k|c_j)$ into account to estimate the probability of surface form c_j describing entity e_k. The logarithm is applied to attain slightly better result values.

Similar to the feature above, the *Term Evidence* feature considers probabilities of context words co-occurring with an entity candidate. For instance, the context word "disease" is an indicator of entity "Influenza" being correct. The term $p(c_j|e_i)$ denotes the probability of context word $c_j \in c_t^\lambda$ co-occurring with entity e_i. Overall, we sum up the probabilities of all context words: $\sum_j^W p(c_j|e_i)$, with $W = |c_t^\lambda|$.

To determine the entity-entity and entity-term distributions we again apply the PLDA approach [10, 14].

3.3 Federated Entity Disambiguation

In the following we present a federated entity disambiguation approach that uses both entity contexts. If an entity-centric or document-centric KB does not provide entity-relevant information, which is more likely in a specialized domain, a federated approach may still retrieve correct disambiguation results. Basically, we rerank disambiguated entities located in the result lists R_l^{ent} and R_l^{doc} of our entity-centric and document-centric disambiguation algorithms by means of LTR which serves as supervised ensemble ranker. The variables ent and doc denote the type of the KB and parameter l denotes the length of the respective approach's result list.

Overall, we compute a new score $S_{e_i}^{com}$ for every entity located in R_l^{ent} and R_l^{doc} and create a new result list. Therefore we first define an entity set M that contains all disambiguated entities of R_l^{ent} and R_l^{doc}. Further, we compute the final score $S_{e_i}^{com}$:

$$S_{e_i}^{com} = w^\intercal f(e_i), \text{with } e_i \in M \tag{8}$$

Similar to Equation 5, variable w denotes the weight vector of our feature set F_{com} and function $f(e_i)$ returns a vector containing the feature values of entity e_i.

Our first two features represent the entity scores $S_{e_i}^{ent}$, $S_{e_i}^{doc}$ attained with our entity-centric and document-centric disambiguation approaches (cf. Equation 5 and 6). Our third feature describes the entity score attained with the combined feature set of entity-centric and document-centric disambiguation. More specific, we compute the linear combination of the weighted feature set comprising the entity-centric feature set F_{ent} and the document-centric classification feature (used in Equation 6). The weights of the corresponding weight vector to compute this feature score are learned in a preprocessing step. Our last two features describe the probability of the entity-centric or document-centric approach retrieving a correct result given the biomedical subdomain of entity e_i. An entity may belong to one of five subdomains as given by our corpus (cf. Section 4). We compute the probabilities by analyzing the results of our approaches.

Overall, we use the top 50 entities of the entity-centric and document-centric algorithms as input entities to provide a good entity repertory for the federated approach.

4 Data Set

To evaluate our properties we have chosen the CALBC (Collaborative Annotation of a Large Biomedical Corpus) data set, a biomedical domain specific KB representing a very large, community-wide shared, silver standard text corpus annotated with biomedical entity references [8]. Overall, we applied the CALBC due to the following reasons:

- In contrast to gold standard corpora like the BioCreative (II) corpora[1], CALBC provides a huge set of annotations which perfectly suit for our evaluation purpose in terms of quantity (24,447 annotations in Biocreative II versus ≈120M annotations in CALBC). It is noted that despite some annotations might be erroneous the corpus most likely serves as predictive surrogate for a gold standard corpora [8].
- It already represents a document-centric KB comprising biomedical documents annotated with biomedical entities, which mostly can be linked to the LOD cloud.

[1] `http://www.biocreative.org/news/biocreative-ii/`

Basically, the data set is released in 3 differently sized corpora: small, big and pilot. For our evaluations we use the small (CALBCSmall, 174.999 documents) and the big (CALBCBig, 714.282 documents) corpus, which mainly differ in the number of available documents. All CALBC documents cover Medline abstracts of the "Immunology" domain, a reasonably broad topic within the biomedical domain. Overall, the set of annotated entities amounts to ≈500.000 distinct biomedical entities overall, compared to ≈100.000 biomedical entities covered by DBpedia [19]. These referenced entities are categorized into four main classes (subdomains) namely, Protein and Genes, Chemicals, Diseases and Disorders as well as Living Beings. All these entities are separated in different namespaces. Due to resources from some of the namespaces are not publicly available we restricted the data set to the available data sets, namely using the namespaces UMLS[2], Uniprot[3], Disease (is a subset of UMLS), EntrezGene[4] and Chemlist[5]. Despite this restriction we still cover the majority of the annotated entities (≈90%) in the corpus. With these entities constituting our sample space, we are able to generate an entity-centric knowledge base by gathering information from LOD repositories. For each user annotation we are able to create a link of the respective RDF resource. To create a KB entry we extract labels, available synonyms, descriptions and functional information. All LOD data sets also provide its own specific properties (e.g. taxonomies) which may be used to enrich the KBs but cannot be exploited across all entities. Several domain- and repository-specific information are stored in our KB but are not used by our disambiguation system so far. Table 2 depicts important statistics of our data sets.

Table 2: Data set statistics

	CALBCSmall	**CALBCBig**
Documents	174.999	714.282
Surface Forms	2.548.900	10.304.172
Unique Surface Forms	50.725	101.439
Annotated Entities	37.309.221	96.526.575
Unique Entities	453.352	308.644
Namespaces	14	16

In CALBC, surface forms are linked to 9 entities on average due to a comprehensive classification system. Thus, we accept several valid entities per surface form.

5 Evaluation

Our approaches are implemented in Java with all queries being executed with Apache Lucene 4.8[6]. For the LTR algorithm we chose Sofia-ml[7], a machine learning framework providing algorithms for massive data sets [7]. These algorithm are mainly embedded in our publicly available disambiguation system *DoSeR*[8] (**D**isambiguation **o**f **Se**mantic **R**esources) which is being developed continuously.

First, we investigate the influence of the entity context onto disambiguation accuracy as well as how different scales of user data affect the results (section 5.2). Second,

[2] http://www.nlm.nih.gov/research/umls/
[3] http://www.uniprot.org
[4] http://www.ncbi.nlm.nih.gov/gene
[5] http://www.cas.org/content/regulated-chemicals
[6] http://lucene.apache.org/
[7] http://code.google.com/p/sofia-ml/
[8] http://purl.org/eexcess/components/research/doser

we evaluate how entity context and user data influences the accuracy with large-scale and heterogeneous KBs (section 5.3). Third, we analyze how disambiguation results evolve after adding different degrees of erroneous user data (section 5.4). The small data set is used for all evaluations and the big data set serves for scalability experiments. We note that our intention was not to compare our approach with other approaches: most publicly available biomedical entity annotators do not return a ranked list (e.g. NCBO annotator[9]), which is a key factor in our evaluation. Instead, the major focus in our work lies on evaluating special domain properties.

We report a set of comprehensive and established measures, comprising mean reciprocal rank (MRR), recall and mean average precision (MAP), which are averaged over 5-fold cross validation runs. The reciprocal rank is the multiplicative inverse of the rank of the first correct result in the result. Average precision denotes the average of all precision@n values of a single disambiguation task. A precision@n value is computed at every correct hit n in the result set [11].

5.1 Basic Parameter Settings

Due to an enormous amount of analyzed parameter combinations, we will only present the most important ones. The context length affects the number of words in both directions, before and after the corresponding surface form. We use a context length of 50 words due to more words worsen the results in all experiments. By using Lucene's TF-IDF score, it must be noted that Lucene's default TF-IDF score also takes internal parameters like term boosting and coordination factor into account. Entity-centric disambiguation always uses fuzzy queries to query the entity mentions and term queries to query the surrounding context. Fuzzy queries match terms with a max. edit distance of 2. Document-centric disambiguation always uses term queries for entity mentions and context queries. When using the document-centric KB, we choose $\tau = 1500$, with τ denoting the amount of documents used for classification. Our result list is trimmed to 10 entities per query to provide a good relation between recall and precision.

5.2 Entity Context and User Data

In this experiment we investigate the influence and effects of the entity context (entity-centric vs. document-centric KB) onto disambiguation accuracy. Furthermore, we use different scales of user data and investigate its effect on the results. We performed the evaluations with different fractions of user data whereby 100% states that all available annotations are used. For all fractions all models were reconstructed accordingly.

Table 3 shows an overview of the results attained by different algorithm combinations with various user data fractions. For a better estimation we can say that 1% of user data corresponds to 1 annotation per entity on average. We compare entity-centric disambiguation (EC), document-centric disambiguation (DC) and the federated disambiguation approach while user data must be available for document-centric and federated disambiguation. Figure 3 shows the MRR and recall of our approaches. To maintain clarity we omit MAP values in this graph. We note that the plot's x-axis starts

[9] `http://bioportal.bioontology.org/annotator`

Table 3: Disambiguation accuracy (MRR, Recall and MAP) of entity-centric, document-centric and federated disambiguation with various amount of user data.

	MRR					**Recall**					**MAP**				
UserData in %	100	20	1	0.1	0	100	20	1	0.1	0	100	20	1	0.1	0
EC	88.0	85.5	70.2	44.7	**36.7**	**76.7**	**74.2**	56.2	29.9	**25.3**	70.7	68.1	**50.9**	28.4	**25.7**
DC	75.5	75.6	71.9	57.1	-	71.7	71.8	**58.9**	**42.2**	-	59.5	59.5	47.8	**33.7**	-
Federated	**92.7**	**92.3**	**73.9**	**58.5**	-	71.8	71.6	58.0	37.3	-	**70.9**	**68.5**	50.8	27.9	-

at 0.1% due to its logarithmic scale to improve visualization and its necessity of user data for document-centric and federated disambiguation.

Assuming that a high amount of user data is available (all annotations in CALBC), entity-centric disambiguation attains a high MRR (88.0%) and recall (76.7%) and significantly outperforms the document-centric approach in all measures. Analyzing the results of the federated approach shows a (significant) increase of the MRR of 4% in contrast to the entity-centric approach considering all available user data. A MRR of ≈93% shows a high level of reliability in terms of ranking a correct entity on top. In contrast, the high recall values (76%) provided by the entity-centric approach are not transfered. Instead, the federated approach attains similar results as provided by the document-centric approach (71%). We assume that optimizing our LTR weights w.r.t recall and using additional features may overcome this deficit. The MAP values of the federated approach are slightly decreased compared to the entity-centric approach. Map values of 70% are decent regarding the number of correct results per surface form (depending on the use case).

Analyzing Figure 3 shows that the amount of user data strongly influences MRR and recall of entity-centric and document-centric disambiguation. While the entity-centric approach significantly outperforms the document-centric approach if enough user data is available, we note reverse results if the amount of user data (significantly) decreases. The less user data available, the higher the advance of the document-centric approach. This is explicable by the increasing dependency of the entity-centric approach on KB quality and availability of exploitable features across entities.

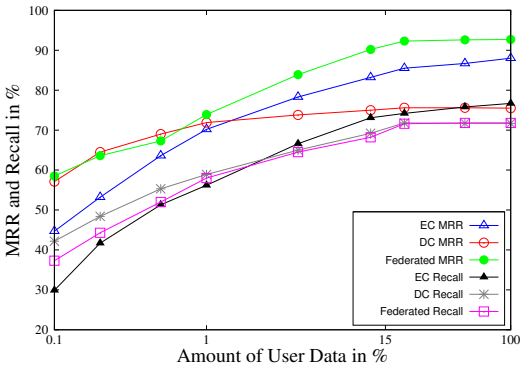

Fig. 3: Results of entity-centric, document-centric and federated disambiguation with various amount of user data.

In summary, we state that neither entity-centric nor document-centric disambiguation attains the best results with all configurations. The choice of entity context that is used to attain the best results strongly depends on the amount of user data. Additionally, the federated approach attains an excellent MRR if enough user data is available.

5.3 Knowledge Base Size and Heterogeneity

In the following we analyze how entity context and user data influence the results when the size and/or heterogeneity of the KBs is increased. We extend our entity-centric KBs KB_{ent} and $KB_{\text{ent/ua/sb}}$ with additional entities. KB_{ent} denotes an entity-centric KB without user data information and $KB_{\text{ent/ua/sb}}$ denotes the enrichment of the entity-centric KB with user data information (ua) of CALBCSmall (s), CALBCBig (b) or both (sb). The set of additional entities comprises all entities belonging to UMLS, Uniprot and/or DBpedia. The document-centric KB is enriched with the CALBCBig data set (intra-specific) and/or Wikipedia pages (inter-specific).

Table 4 shows an overview of the results before and after extending the KBs. The column $Change$ takes the average change of the measures MRR, recall and MAP in $\%$. The entity-centric approach attains worse results after increasing the amount of entities when no user data is available. Additionally, increasing the domain heterogeneity by adding DBpedia entities significantly worsens the results with a decrease of 33 percent (with DBpedia only), respectively 40 percent (with DBpedia, UMLS and Uniprot) on average. An entity-centric disambiguation that applies features derived from annotated documents significantly improves the robustness against an increase of entities and heterogeneity by one third. The usage of additionally mined features from CALBCBig increases these results by 3%. The accuracy drop by about 30% with a KB containing DBpedia, UMLS and Uniprot remains constant. All in all, disambiguation with an entity-centric KB is not robust against large-scale and heterogeneous KBs due to our feature set still does not provide enough evidence to overcome these limitations. It is an open question whether there exist features that suppress these negative effects.

When using a document-centric KB, the results do not suffer when adding more documents with biomedical content. This can be explained with the document increase does not influence the classification (cf. Section 3.1). Instead, the retrieval step has a wider range of documents to choose for the classification step. Selecting other documents has no negative effect on the documents' spectrum of annotated entities. An inter-specific domain extension with CALBC and Wikipedia documents causes a decrease of

Table 4: Results after increasing our KB with various corpora in the biomedical domain

Settings	Integrated KBs	MRR in %	Recall in %	MAP in %	#Ent/#Docs	Change in %
$KB_{\text{ent, intra}}$	-	36.7	25.3	25.7	265.532	-
$KB_{\text{ent, intra}}$	UMLS, Uniprot	30.9	20.4	19.5	32.407.960	-19.3
$KB_{\text{ent, inter}}$	DBpedia	25.6	17.7	18.3	4.643.509	-33.2
$KB_{\text{ent, inter}}$	UMLS, Uniprot, DBpedia	22.9	14.0	15.4	36.785.937	-40.4
$KB_{\text{ent/ua/s, intra}}$	-	88.0	76.7	70.7	265.532	-
$KB_{\text{ent/ua/sb, intra}}$	-	90.5	79.2	73.2	265.532	+3.1
$KB_{\text{ent/ua/s, intra}}$	UMLS, Uniprot	78.0	66.6	60.9	32.407.960	-9.9
$KB_{\text{ent/ua/s, inter}}$	UMLS, Uniprot, DBpedia	60.3	55.9	50.1	36.785.937	-29.3
$KB_{\text{ent/ua/sb, inter}}$	UMLS, Uniprot, DBpedia	62.7	58.0	52.4	36.785.937	-26.4
$KB_{\text{doc, intra}}$	-	75.5	71.7	59.5	174.999	-
$KB_{\text{doc, intra}}$	CALBCBig	76.0	72.2	60.1	889.282	+0.1
$KB_{\text{doc, inter}}$	CALBCBig, Wiki	67.3	65.0	50.8	4.267.259	-11.4
$KB_{\text{federated, intra}}$	-	92.7	71.8	70.9	440.531	-
$KB_{\text{federated, intra}}$	CALBCBig, UMLS, Uniprot	81.9	65.9	61.5	33.297.242	-11.1
$KB_{\text{federated, inter}}$	CALBCBig, UMLS, Uniprot, DBpedia, Wiki	75.7	60.1	51.6	37.675.219	-20.4

the disambiguation results (11%). However, document-centric disambiguation is more robust against an inter-specific domain extension than entity-centric disambiguation.

The federated approach mitigates the accuracy decrease, compared to the entity-centric approach. With the document-centric approach being robust against the document count, the accuracy decrease after increasing heterogeneity and entity/document count stays small.

In summary, disambiguating biomedical entities from the LOD cloud with a document-centric approach is more robust against large-scale and heterogeneous KBs than entity-centric disambiguation. The results recommend to use a federated approach to yield the advantages of both approaches (result and robustness against large-scale KBs).

5.4 Noisy User Data

Available user data may contain errors caused by missing knowledge, validation etc. While the original CALBC may contain erroneous annotations due to constituting a silver-standard corpus we investigate how additional noise in its annotations influence results attained with the entity-centric, document-centric and federated disambiguation approach. We compare a user model created from the original annotations (as given by CALBC) with user models with different degrees of additional annotation errors. Prior research has already investigated the influence of noisy user data on LTR models, but the effects on disambiguation results are unknown. We modified available CALBC annotations and recreated our KBs as well as LTR models. Therefore we selected an annotation to be wrong with probability p. Instead of exchanging this entity annotation with a randomly selected entity annotation, we simulated user behavior by choosing a wrong entity from the result list of a conducted disambiguation task (entity-centric disambiguation) on the annotation's surface form. Choosing a wrong entity at the top of the result list should be more likely than choosing an entity from the end. We modeled this event with a Gaussian distributed random variable $X \sim \mathcal{N}(1, 10)$ which yields positive values only. We exchanged the correct annotation with the wrong result that was selected by the random variable. We modified the CALBC annotations with varying degree of noise. Figure 4 shows the evaluation results from 0% additional noise (as given by CALBC) to 100% noise (all annotations are wrong) attained with an entity-centric, document-centric and federated approach. In the

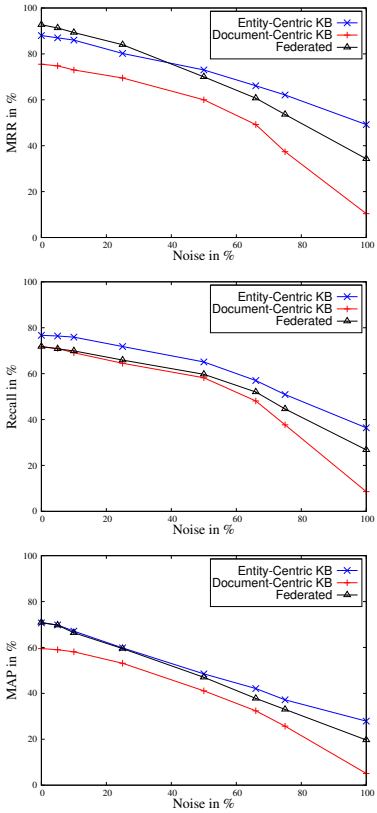

Fig. 4: Influence of noise in user data on disambiguation results.

following, we focus on the results with a 25% noise rate. The MRR of the entity-centric and document-centric approach provides a slight decrease of 10% with a noise rate of 25%. The federated approach tops the single approaches as long as the noise stays below 33%. In all approaches, the recall decrease is about 5% with 25% noise. Basically, the recall values stay high as long as the noise rate does not exceed 66%. It another story for the MAP values which continuously decrease almost linearly from 0 to 100% noise with the entity-centric and federated approach. However, a decrease of up to 12% with 25% noise in all approaches shows that the MAP results are influenced by noisy user data.

In summary, all approaches are robust against little noise in the user data. Assuming that the amount of erroneously annotated data is about one third or less, we note that all disambiguation approaches are robust and still provide fairly satisfying results.

6 Related Work

One of the first works to disambiguate general knowledge entities (e.g. Wikipedia) defines a similarity measure to compute the cosine similarity between the text around the surface form and the referent entity candidates' Wikipedia page [1]. At the same time Cucerzan et al. introduced topical coherence for entity disambiguation [2]. The authors use the referent entity candidate and other entities within the same context to compute topical coherence by analyzing the overlap of categories and incoming links in Wikipedia. Several works use topical coherence and context similarity to improve disambiguation [9, 15, 18]. All these works exploit various Wikipedia features (e.g. categories). There are some more generic approaches that can be easily applied to other KBs. Subsequent work incorporate more information to improve entity context similarity comparison by exploring query expansion [3]. Another work proposes a generative entity-mention model which, similar to our work, consists of an underlying entity popularity model, entity name model and entity context model [4]. Some other works propose generic, generative topic-models for entity disambiguation which exploit context compatibility and topic coherence [5, 17]. Almost all works address algorithm improvements but do not investigate the requirements to adapt the results to other domains.

Biomedical entity disambiguation has also attained much attention in research in the last decade [24]. For instance, Wang et. al classify relations between entities for biomedical entity disambiguation [21]. Biomedical entities can also be disambiguated with the help of species disambiguation. Wang et al. [22] apply language parsers for species disambiguation and attain promising results. Zwicklbauer et al. [23] compared document-centric and entity-centric KBs with a search-based algorithm. The authors report very strong results with document-centric KBs in the biomedical domain.

In terms of entity context, several works use intensional entity descriptions provided by high-quality KBs like DBpedia [12, 18, 20], which are similar to our entity-centric approach. Some other works store Wikipedia documents as a whole in a KB to describe entities [17], but exploit that Wikipedia articles describe one specific entity. In contrast, the authors of [5] use a document-centric KB containing arbitrary entity-annotated documents. This generative approach jointly models context compatibility, topic coherence and its correlation, while our algorithm constitutes a retrieval-based approach.

7 Conclusion and Future Work

We provide a systematic evaluation of biomedical entity disambiguation with respect to three major properties of specialized domain disambiguation systems, namely the entity context, user data and the quantity and heterogeneity of entities to disambiguate. Our evaluation reveals that the choice of entity context that is used to attain the best disambiguation results strongly depends on the amount of available user data. In this context, we indicate that the performance decrease with large-scale and heterogeneous KBs strongly depends on the underlying entity context. Additionally, we show that disambiguation results are robust against a moderate amount of noise in user data. Finally, we suggest to use a federated approach of different entity contexts to improve the reciprocal rank and to increase the robustness against large-scale and heterogeneous KBs.

In summary, we state that disambiguation systems must be carefully adapted when expanding their KBs with special domain entities. An analysis of the underlying data set is strongly required to spot the potential problem areas and integrate the appropriate approaches. In this context our future work includes the design of a model that automatically analyzes the underlying KB and chooses the best disambiguation settings.

8 Acknowledgments

The presented work was developed within the EEXCESS project funded by the European Union Seventh Framework Programme FP7/2007-2013 under grant agreement number 600601.

References

1. Bunescu, R., Pasca, M.: Using Encyclopedic Knowledge for Named Entity Disambiguation. In: Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06), Trento, Italy. pp. 9–16 (2006)
2. Cucerzan, S.: Large-scale named entity disambiguation based on Wikipedia data. In: In Proc. 2007 Joint Conference on EMNLP and CoNLL. pp. 708–716. Association for Computational Linguistics, Prague, Czech Republic (June 2007)
3. Gottipati, S., Jiang, J.: Linking entities to a knowledge base with query expansion. In: Proc. of the Conf. on Empirical Methods in Natural Language Processing. pp. 804–813. EMNLP '11, ACL, Stroudsburg, PA, USA (2011)
4. Han, X., Sun, L.: A generative entity-mention model for linking entities with knowledge base. In: Proc. of the 49th Annual Meeting of the Association for Computational Linguistics. pp. 945–954. HLT '11, ACL, Stroudsburg, PA, USA (2011)
5. Han, X., Sun, L.: An entity-topic model for entity linking. In: Proc. of the 2012 Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. pp. 105–115. EMNLP-CoNLL '12, ACL, Stroudsburg, PA, USA (2012)
6. Jaro, M.A.: Advances in record-linkage methodology as applied to matching the 1985 census of Tampa, Florida. Journal of the American Statistical Association 84(406), 414–420 (1989)
7. Joachims, T.: Optimizing search engines using clickthrough data. In: Proc. of the eighth ACM SIGKDD international Conf. on Knowledge Discovery and Data Mining. pp. 133–142. KDD '02, ACM, New York, NY, USA (2002)

8. Kafkas, S., Lewin, I., Milward, D., van Mulligen, E., Kors, J., Hahn, U., Rebholz-Schuhmann, D.: Calbc: Releasing the final corpora. In: Proc. of the Eight International Conf. on Language Resources and Evaluation (LREC'12). Istanbul, Turkey (May 2012)
9. Kataria, S.S., Kumar, K.S., Rastogi, R.R., Sen, P., Sengamedu, S.H.: Entity disambiguation with hierarchical topic models. In: Proc. of the 17th ACM SIGKDD international Conf. on Knowledge Discovery and Data Mining. pp. 1037–1045. KDD '11, ACM, NY, USA (2011)
10. Li, Y., Wang, C., Han, F., Han, J., Roth, D., Yan, X.: Mining evidences for named entity disambiguation. In: Proc. of the 19th ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining. pp. 1070–1078. KDD '13, ACM, New York, NY, USA (2013)
11. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York, NY, USA (2008)
12. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: Dbpedia spotlight: shedding light on the web of documents. In: Proc. of the 7th International Conf. on Semantic Systems. pp. 1–8. I-Semantics '11, ACM, New York, NY, USA (2011)
13. Ogden, C., Richards, I.A.: The meaning of meaning: A study of the influence of language upon thought and of the science of symbolism. 8th ed. 1923. Reprint New York: Harcourt Brace Jovanovich (1923)
14. Ramage, D., Manning, C.D., Dumais, S.: Partially labeled topic models for interpretable text mining. In: Proc. of the 17th ACM SIGKDD International Conf. on Knowledge Discovery and Data Mining. pp. 457–465. KDD '11, ACM, New York, NY, USA (2011)
15. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to wikipedia. In: Proc. of the Annual Meeting of the Association of Computational Linguistics (2011)
16. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1. pp. 448–453. IJCAI'95, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1995)
17. Sen, P.: Collective context-aware topic models for entity disambiguation. In: Proc. of the 21st International Conf. on World Wide Web. pp. 729–738. WWW '12, ACM, New York, NY, USA (2012)
18. Shen, W., Wang, J., Luo, P., Wang, M.: Linden: linking named entities with knowledge base via semantic knowledge. In: Proc. of the 21st International Conf. on World Wide Web. pp. 449–458. WWW '12, ACM, New York, NY, USA (2012)
19. Tian, L., Zhang, W., Bikakis, A., Wang, H., Yu, Y., Ni, Y., Cao, F.: Medetect: A lod-based system for collective entity annotation in biomedicine. In: Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on. vol. 1, pp. 233–240. IEEE (2013)
20. Usbeck, R., Ngomo, A.C.N., Röder, M., Gerber, D., Coelho, S.A., Auer, S., Both, A.: Agdistis-graph-based disambiguation of named entities using linked data. In: The Semantic Web–ISWC 2014, pp. 457–471. Springer (2014)
21. Wang, X., Tsujii, J., Ananiadou, S.: Classifying relations for biomedical named entity disambiguation. In: Proc. of the 2009 Conf. on Empirical Methods in Natural Language Processing: Volume 3. pp. 1513–1522. EMNLP '09, ACL, Stroudsburg, PA, USA (2009)
22. Wang, X., Tsujii, J., Ananiadou, S.: Disambiguating the species of biomedical named entities using natural language parsers. Bioinformatics 26(5), 661–667 (2010)
23. Zwicklbauer, S., Seifert, C., Granitzer, M.: Do we need entity-centric knowledge bases for entity disambiguation? In: Proc. of the 13th International Conference on Knowledge Management and Knowledge Technologies. pp. 4:1–4:8. i-Know '13, ACM, NY, USA (2013)
24. Zwicklbauer, S., Seifert, C., Granitzer, M.: Linking biomedical data to the cloud. In: Smart Health, pp. 209–235. Springer (2015)