CESI: Canonicalizing Open Knowledge Bases using Embeddings and Side Information

Shikhar Vashishth
Indian Institute of Science
Bangalore, India
shikhar@iisc.ac.in

Prince Jain*
Microsoft
Bangalore, India
prince.jain@microsoft.com

Partha Talukdar
Indian Institute of Science
Bangalore, India
ppt@iisc.ac.in

ABSTRACT

Open Information Extraction (OpenIE) methods extract *(noun phrase, relation phrase, noun phrase)* triples from text, resulting in the construction of large Open Knowledge Bases (Open KBs). The noun phrases (NPs) and relation phrases in such Open KBs are not *canonicalized*, leading to the storage of redundant and ambiguous facts. Recent research has posed canonicalization of Open KBs as clustering over *manually-defined* feature spaces. Manual feature engineering is expensive and often sub-optimal. In order to overcome this challenge, we propose Canonicalization using Embeddings and Side Information (CESI) – a novel approach which performs canonicalization over *learned* embeddings of Open KBs. CESI extends recent advances in KB embedding by incorporating relevant NP and relation phrase side information in a principled manner. Through extensive experiments on multiple real-world datasets, we demonstrate CESI's effectiveness.

CCS CONCEPTS

• **Computing methodologies** → *Knowledge representation and reasoning*; *Information extraction*;

KEYWORDS

Canonicalization; Knowledge Graphs; Knowledge Graph Embeddings; Open Knowledge Bases

ACM Reference Format:
Shikhar Vashishth, Prince Jain, and Partha Talukdar. 2018. CESI: Canonicalizing Open Knowledge Bases using Embeddings and Side Information. In *WWW 2018: The 2018 Web Conference, April 23–27, 2018, Lyon, France.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3178876.3186030

1 INTRODUCTION

Recent research has resulted in the development of several large *Ontological* Knowledge Bases (KBs), examples include DBpedia [1], YAGO [36], and Freebase [4]. These KBs are called ontological as the knowledge captured by them conform to a fixed ontology, i.e., pre-specified Categories (e.g., *person, city*) and Relations (e.g., *mayorOfCity(Person, City)*). Construction of such ontological KBs require significant human supervision. Moreover, due to the need for pre-specification of the ontology, such KB construction methods can't be quickly adapted to new domains and corpora. While other ontological KB construction approaches such as NELL [23] learn from limited human supervision, they still suffers from the quick adaptation bottleneck.

In contrast, Open Information Extraction (OpenIE) methods need neither supervision nor any pre-specified ontology. Given unstructured text documents, OpenIE methods readily extract triples of the form *(noun phrase, relation phrase, noun phrase)* from them, resulting in the development of large Open Knowledge Bases (Open KBs). Examples of Open KBs include TextRunner [3], ReVerb [12], and OLLIE [8, 21, 33]. While this makes OpenIE methods highly adaptable, they suffer from the following shortcoming: unlike Ontological KBs, the Noun Phrases (NPs) and relation phrases in Open KBs are not *canonicalized*. This results in storage of redundant and ambiguous facts.

Let us explain the need for canonicalization through a concrete example. Please consider the two sentences below.

Barack Obama was the president of US.
Obama was born in Honolulu.

Given the two sentences above, an OpenIE method may extract the two triples below and store them in an Open KB.

(Barack Obama, was president of, US)
(Obama, born in, Honolulu)

Unfortunately, neither such OpenIE methods nor the associated Open KBs have any knowledge that both *Barack Obama* and *Obama* refer to the same person. This can be a significant problem as Open KBs will not return all the facts associated with *Barack Obama* on querying for it. Such KBs will also contain redundant facts, which is undesirable. Thus, there is an urgent need to *canonicalize* noun phrases (NPs) and relations in Open KBs.

In spite of its importance, canonicalization of Open KBs is a relatively unexplored problem. In [14], canonicalization of Open KBs is posed as a clustering problem over *manually* defined feature representations. Given the costs and sub-optimality involved with manual feature engineering, and inspired by recent advances in knowledge base embedding [5, 25], we pose canonicalization of Open KBs as a clustering over *automatically learned* embeddings. We make the following contributions in this paper.

- We propose Canonicalization using Embeddings and Side Information (CESI), a novel method for canonicalizing Open KBs using learned embeddings. To the best of our knowledge, this is the first approach to use learned embeddings and side information for canonicalizing an Open KB.

*Research carried out while at the Indian Institute of Science, Bangalore.

WWW 2018, April 23–27, 2018, Lyon, France

ACM ISBN 978-1-4503-5639-8/18/04.
https://doi.org/10.1145/3178876.3186030

- CESI models the problem of noun phrase (NP) and relation phrase canonicalization *jointly* using relevant side information in a principled manner. This is unlike prior approaches where NP and relation phrase canonicalization were performed sequentially.
- We build and experiment with ReVerb45K, a new dataset for Open KB canonicalization. ReVerb45K consists of 20x more NPs than the previous biggest dataset for this task. Through extensive experiments on this and other real-world datasets, we demonstrate CESI's effectiveness (Section 7).

CESI's source code and datasets used in the paper are available at https://github.com/malllabiisc/cesi.

2 RELATED WORK

Entity Linking: One traditional approach to canonicalizing noun phrases is to map them to an existing KB such as Wikipedia or Freebase. This problem is known as Entity Linking (EL) or Named Entity Disambiguation (NED). Most approaches generate a list of candidate entities for each NP and re-rank them using machine learning techniques. Entity linking has been an active area of research in the NLP community [19, 32, 39]. A major problem with these kind of approaches is that many NPs may refer to new and emerging entities which may not exist in KBs. One approach to resolve these noun phrases is to map them to NIL or an OOKB (Out of Knowledge Base) entity, but the problem still remains as to how to cluster these NIL mentions. Although entity linking is not the best approach to NP canonicalization, we still leverage signals from entity linking systems for improved canonicalization in CESI.

Canonicalization in Ontological KBs: Concept Resolver [17] is used for clustering NP mentions in NELL [23]. It makes "one sense per category" assumption which states that a noun phrase can refer to at most one concept in each category of NELL's ontology. For example, the noun phrase "Apple" can either refer to a company or a fruit, but it can refer to only one company and only one fruit. Another related problem to NP canonicalization is Knowledge Graph Identification [31], where given a noisy extraction graph, the task is to produce a consistent Knowledge Graph (KG) by performing entity resolution, entity classification and link prediction jointly. Pujara et al. [31] incorporate information from multiple extraction sources and use ontological information to infer the most probable knowledge graph using probabilistic soft logic (PSL) [6]. However, both of these approaches require additional information in the form of an ontology of relations, which is not available in the Open KB setting.

Relation Taxonomy Induction: SICTF [27] tries to learn relation schemas for different OpenIE relations. It is built up on RESCAL [26], and uses tensor factorization methods to cluster noun phrases into *categories* (such as "person", "disease", etc.). We, however, are interested in clustering noun phrases into entities.

There has been relatively less work on the task of relation phrase canonicalization. Some of the early works include DIRT [18], which proposes an unsupervised method for discovering inference rules of the form "*X is the author of Y* $\approx$ *X wrote Y*" using paths in dependency trees; and the PATTY system [24], which tries to learn subsumption rules among relations (such as *son-of* $\subset$ *child-of*) using techniques based on frequent itemset mining. These approaches are more focused on finding a taxonomy of relation phrases, while we are looking at finding equivalence between relation phrases.

Knowledge Base Embedding: KB embedding techniques such as TransE [5], HolE [25] try to learn vector space embeddings for entities and relations present in a KB. TransE makes the assumption that for any ⟨*subject, relation, object*⟩ triple, the relation vector is a translation from the subject vector to the object vector. HolE, on the other hand, uses non-linear operators to model a triple. These embedding methods have been successfully applied for the task of link prediction in KBs. In this work, we build up on HolE while exploiting relevant side information for the task of Open KB canonicalization. We note that, even though KB embedding techniques like HolE have been applied to ontological KBs, CESI might be the first attempt to use them in the context of Open KBs.

Canonicalizing Open KBs: The RESOLVER system [42] uses string similarity based features to cluster phrases in TextRunner [3] triples. String similarity features, although being effective, fail to handle synonymous phrases which have completely different surface forms, such as *Myopia* and *Near-sightedness*.

KB-Unify [10] addresses the problem of unifying multiple Ontological and Open KBs into one KB. However, KB-Unify requires a pre-determined sense inventory which is not available in the setting CESI operates.

The most closely related work to ours is [14]. They perform NP canonicalization by performing Hierarchical Agglomerative Clustering (HAC) [38] over manually-defined feature spaces, and subsequently perform relation phrase clustering by using the AMIE algorithm [15]. CESI significantly outperforms this prior method (Section 7).

3 PROPOSED APPROACH: CESI

Overall architecture and dataflow of CESI is shown in Figure 1. The input to CESI is an un-canonicalized Open Knowledge Base (KB) with source information for each triple. The output is a list of canonicalized noun and relation phrases, which can be used to identify equivalent entities and relations or canonicalize the KB. CESI achieves this through its three step procedure:

(1) **Side Information Acquisition:** The goal of this step is to gather various NP and relation phrase side information for each triple in the input by running several standard algorithms on the source text of the triples. More details can be found in Section 4.
(2) **Embedding NP and Relation Phrases:** In this step, CESI learns specialized vector embeddings for all NPs and relation phrases in the input by making principled use of side information available from the previous step.
(3) **Clustering Embeddings and Canonicalization:** Goal of this step is to cluster the NPs and relation phrases on the basis of their distance in the embedding space. Each cluster represents a specific entity or relation. Based on certain relevant heuristics, we assign a representative to each NP and relation phrase cluster.

Details of different steps of CESI are described next.

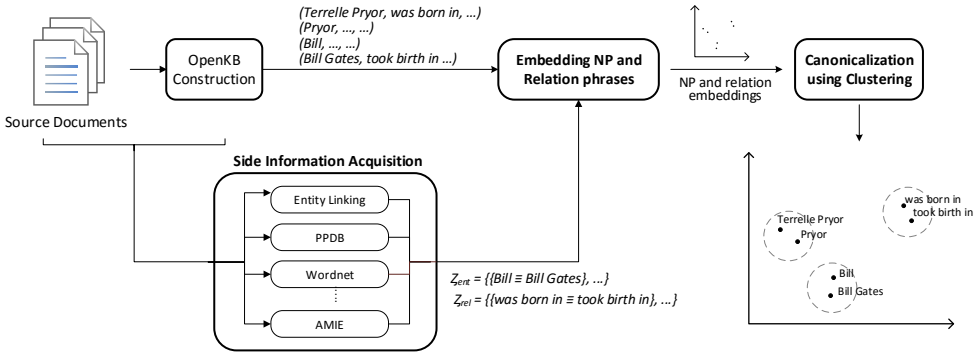


Figure 1: Overview of CESI. CESI first acquires side information of noun and relation phrases of Open KB triples. In the second step, it learns embeddings of these NPs and relation phrases while utilizing the side information obtained in previous step. In the third step, CESI performs clustering over the learned embeddings to canonicalize NP and relation phrases. Please see Section 3 for more details.

4 SIDE INFORMATION ACQUISITION

Noun and relation phrases in Open KBs often have relevant side information in the form of useful context in the documents from which the triples were extracted. Sometimes, such information may also be present in other related KBs. Previous Open KB canonicalization methods [14] ignored such available side information and performed canonicalization in isolation focusing only on the Open KB triples. CESI attempts to exploit such side information to further improve the performance on this problem. In CESI, we make use of five types of NP side information to get equivalence relations of the form $e_1 \equiv e_2$ between two entities e_1 and e_2. Similarly, relation phrase side information is used to derive relation equivalence, $r_1 \equiv r_2$. All equivalences are used as soft constraints in later steps of CESI (details in Section 5).

4.1 Noun Phrase side Information

In the present version of CESI, we make use of the following five types of NP side information:

(1) **Entity Linking**: Given unstructured text, entity linking algorithms identify entity mentions and link them to Ontological KBs such as Wikipedia, Freebase etc. We make use of Stanford CoreNLP entity linker which is based on [35] for getting NP to Wikipedia entity linking. Roughly, in about 30% cases, we get this information for NPs. If two NPs are linked to the same Wikipedia entity, we assume them to be equivalent as per this information. For example, *US* and *America* can get linked to the same Wikipedia entity *United_States*.

(2) **PPDB Information**: We make use of PPDB 2.0 [29], a large collection of paraphrases in English, for identifying equivalence relation among NPs. We first extracted high confidence paraphrases from the dataset while removing duplicates. Then, using union-find, we clustered all the equivalent phrases and randomly assigned a representative to each cluster. Using an index created over the obtained clusters, we find cluster representative for each NP. If two NPs have the same cluster representative then they are considered to be equivalent. NPs not present in the dataset are skipped. This information helps us identifying equivalence between NPs such as *management* and *administration*.

(3) **WordNet with Word-sense Disambiguation**: Using word-sense disambiguation [2] with Wordnet [22], we identify possible synsets for a given NP. If two NPs share a common synset, then they are marked as similar as per this side information. For example, *picture* and *image* can get linked to the same synset *visualize.v.01*.

(4) **IDF Token Overlap**: NPs sharing infrequent terms give a strong indication of them referring to the same entity. For example, it is very likely for *Warren Buffett* and *Buffett* to refer to the same person. In [14], IDF token overlap was found to be the most effective feature for canonicalization. We assign a score for every pair of NPs based on the standard IDF formula:

$$score_{idf}(n, n') = \frac{\sum_{x \in w(n) \cap w(n')} \log (1 + f(x))^{-1}}{\sum_{x \in w(n) \cup w(n')} \log (1 + f(x))^{-1}}$$

Here, $w(\cdot)$ for a given NP returns the set of its terms, excluding stop words. $f(\cdot)$ returns the document frequency for a token.

(5) **Morph Normalization**: We make use of multiple morphological normalization operations like tense removal, pluralization, capitalization and others as used in [12] for finding out equivalent NPs. We show in Section 8.2 that this information helps in improving performance.

4.2 Relation Phrase Side Information

Similar to noun phrases, we make use of PPDB and WordNet side information for relation phrase canonicalization as well. Apart from these, we use the following two additional types of side information involving relation phrases.

(1) **AMIE Information**: AMIE algorithm [15] tries to learn implication rules between two relations r and r' of the form

$r \Rightarrow r'$. These rules are detected based on statistical rule mining, for more details refer [14]. It declares two relations r and r' to be equivalent if both $r \Rightarrow r'$ and $r' \Rightarrow r$ satisfy support and confidence thresholds. AMIE accepts a semi-canonicalized KB as input, i.e., a KB where NPs are already canonicalized. Since this is not the case with Open KBs, we first canonicalized NPs morphologically and then applied AMIE over the NP-canonicalized KB. We chose morphological normalization for this step as such normalization is available for all NPs, and also because we found this side information to be quite effective in large Open KBs.

(2) **KBP Information**: Given unstructured text, Knowledge Base Population (KBP) systems detect relations between entities and link them to relations in standard KBs. For example, *"Obama was born in Honolulu"* contains *"was born in"* relation between *Obama* and *Honolulu*, which can be linked to *per:city_of_birth* relation in KBs. In CESI, we use Stanford KBP [37] to categorize relations. If two relations fall in the same category, then they are considered equivalent as per this information.

The given list can be further extended based on the availability of other side information. For the experiments in this paper, we have used the above mentioned NP and relation phrase side information. Some of the equivalences derived from different side information might be erroneous, therefore, instead of using them as hard constraints, we try to use them as supplementary information as described in the next section. Even though side information might be available only for a small fraction of NPs and relation phrases, the hypothesis is that it will result in better overall canonicalization. We find this to be true, as shown in Section 8.

5 EMBEDDING NP AND RELATION PHRASES

For learning embeddings of NPs and relation phrases in a given Open KB, CESI optimizes HolE's [25] objective function along with terms for penalizing violation of equivalence conditions from the NP and relation phrase side information. Since the conditions from side information might be spurious, a factor ($\lambda_{\text{ent/rel},\theta}$) is multiplied with each term, which acts as a hyper-parameter and is tuned on a held out validation set. We also keep a constant (λ_{str}) with HolE objective function, to make selective use of structural information from KB for canonicalization. We choose HolE because it is one of the best performing KB embeddings techniques for tasks like link prediction in knowledge graphs. Since KBs store only true triples, we generate negative examples using local closed world heuristic [11]. To keep the rank of true triples higher than the non-existing ones, we use pairwise ranking loss function. The final objective function is described below.

$$
\begin{aligned}
\min_{\Theta} \; & \lambda_{str} \sum_{i \in D_+} \sum_{j \in D_-} \max(0, \gamma + \sigma(\eta_j) - \sigma(\eta_i)) \\
& + \sum_{\theta \in \mathscr{C}_{\text{ent}}} \frac{\lambda_{\text{ent},\theta}}{|\mathcal{Z}_{\text{ent},\theta}|} \sum_{v, v' \in \mathcal{Z}_{\text{ent},\theta}} \|e_v - e_{v'}\|^2 \\
& + \sum_{\phi \in \mathscr{C}_{\text{rel}}} \frac{\lambda_{\text{rel},\phi}}{|\mathcal{Z}_{\text{rel},\phi}|} \sum_{u, u' \in \mathcal{Z}_{\text{rel},\phi}} \|r_u - r_{u'}\|^2 \\
& + \lambda_{\text{reg}} \left(\sum_{v \in V} \|e_v\|^2 + \sum_{r \in R} \|e_r\|^2 \right).
\end{aligned}
$$

The objective function, consists of three main terms, along with one term for regularization. Optimization parameter, $\Theta = \{e_v\}_{v \in V} \cup \{r_u\}_{u \in R}$, is the set of all NP ($e_v$) and relation phrase ($r_u$) d-dimensional embeddings, where, V and R denote the set of all NPs and relation phrases in the input. In the first term, D_+, D_- specify the set of positive and negative examples and $\gamma > 0$ refers to the width of the margin [5]. Further, $\sigma(\cdot)$ denotes the logistic function and for a triple t_i (s, p, o), $\eta_i = r_p^T (e_s \star e_o)$, where $\star : R^d \times R^d \rightarrow R^d$ is the circular correlation operator defined as follows.

$$
[a \star b]_k = \sum_{i=0}^{d-1} a_i b_{(k+i) \bmod d}.
$$

The first index of $(a \star b)$ measures the similarity between a and b, while other indices capture the interaction of features from a and b, in a particular order. Please refer to [25] for more details.

In the second and third terms, $\mathscr{C}_{\text{ent}}$ and $\mathscr{C}_{\text{rel}}$ are the collection of all types of NP and relation side information available from the previous step (Section 4), i.e., $\mathscr{C}_{\text{ent}}$ = {Entity Linking, PPDB, ..} and $\mathscr{C}_{\text{rel}}$ = {AMIE, KBP, ..}. Further, $\lambda_{\text{ent},\theta}$ and $\lambda_{\text{rel},\phi}$ denote the constants associated with entity and relation side information. Their value is tuned using grid search on a held out validation set. The set of all equivalence conditions from a particular side information is denoted by $\mathcal{Z}_{\text{ent},\theta}$ and $\mathcal{Z}_{\text{rel},\phi}$. The rationale behind putting these terms is to allow inclusion of side information while learning embeddings, by enforcing two NPs or relations close together if they are equivalent as per the available side information. Since the side information is available for a fraction of NPs and relation phrases in the input, including these terms in the objective does not slow down the training of embeddings significantly.

The last term adds L2 regularization on the embeddings. All embeddings are initialized by averaging GloVe vectors [30]. We use mini-batch gradient descent for optimization.

6 CLUSTERING EMBEDDINGS AND CANONICALIZATION

CESI clusters NPs and relation phrases by performing Hierarchical Agglomerative Clustering (HAC) using cosine similarity over the embeddings learned in the previous step (Section 5). HAC was preferred over other clustering methods because the number of clusters are not known beforehand. Complete linkage criterion is used for calculating the similarity between intermediate clusters as it gives smaller sized clusters, compared to single and average linkage criterion. This is more reasonable for canonicalization problem,

Datasets	# Gold Entities	#NPs	#Relations	#Triples
Base	150	290	3K	9K
Ambiguous	446	717	11K	37K
ReVerb45K	7.5K	15.5K	22K	45K

Table 1: Details of datasets used. ReVerb45K is the new dataset we propose in this paper. Please see Section 7.1 for details.

where cluster sizes are expected to be small. The threshold value for HAC was chosen based on held out validation dataset.

The time complexity of HAC with complete linkage criterion is $O(n^2)$ [9]. For scaling up CESI to large knowledge graphs, one may go for modern variants of approximate Hierarchical clustering algorithms [16] at the cost of some loss in performance.

Finally, we decide a representative for each NP and relation phrase cluster. For each cluster, we compute a mean of all elements' embeddings weighted by the frequency of occurrence of each element in the input. NP or relation phrase which lies closest to the weighted cluster mean is chosen as the representative of the cluster.

7 EXPERIMENTAL SETUP

7.1 Datasets

Statistics of the three datasets used in the experiments of this paper are summarized in Table 1. We present below brief summary of each dataset.

(1) **Base and Ambiguous Datasets:** We obtained the Base and Ambiguous datasets from the authors of [14]. Base dataset was created by collecting triples containing 150 sampled Freebase entities that appear with at least two aliases in ReVerb Open KB. The same dataset was further enriched with mentions of homonym entities to create the Ambiguous dataset. Please see [14] for more details.

(2) **ReVerb45K:** This is the new Open KB canonicalization dataset we propose in this paper. ReVerb45K is a significantly extended version of the Ambiguous dataset, containing more than 20x NPs. ReVerb45K is constructed by intersecting information from the following three sources: ReVerb Open KB [12], Freebase entity linking information from [13], and Clueweb09 corpus [7]. Firstly, for every triple in ReVerb, we extracted the source text from Clueweb09 corpus from which the triple was generated. In this process, we rejected triples for which we could not find any source text. Then, based on the entity linking information from [13], we linked all subjects and objects of triples to their corresponding Freebase entities. If we could not find high confidence linking information for both subject and object in a triple, then it was rejected. Further, following the dataset construction procedure adopted by [14], we selected triples associated with all Freebase entities with at least two aliases occurring as subject in our dataset. Through these steps, we obtained 45K high-quality triples which we used for evaluation. We call this resulting dataset ReVerb45K.

In contrast to Base and Ambiguous datasets, the number of entities, NPs and relation phrases in ReVerb45K are significantly larger. Please see Table 1 for a detailed comparison. This better mimics real-world KBs which tend to be sparse with very few edges per entity, as also observed by [5].

For getting test and validation set for each dataset, we randomly sampled 20% Freebase entities and called all the triples associated with them as validation set and rest was used as the test set.

7.2 Evaluation Metrics

Following [14], we use macro-, micro- and pairwise metrics for evaluating Open KB canonicalization methods. We briefly describe below these metrics for completeness. In all cases, C denotes the clusters produced by the algorithm to be evaluated, and E denotes the gold standard clusters. In all cases, F1 measure is given as the harmonic mean of precision and recall.

Macro: Macro precision (P_{macro}) is defined as the fraction of pure clusters in C, i.e., clusters in which all the NPs (or relations) are linked to the same gold entity (or relation). Macro recall (R_{macro}) is calculated like macro precision but with the roles of E and C interchanged.

$$P_{\text{macro}}(C,E) = \frac{|\{c \in C : \exists e \in E : e \supseteq c\}|}{|C|}$$
$$R_{\text{macro}}(C,E) = P_{\text{macro}}(E,C)$$

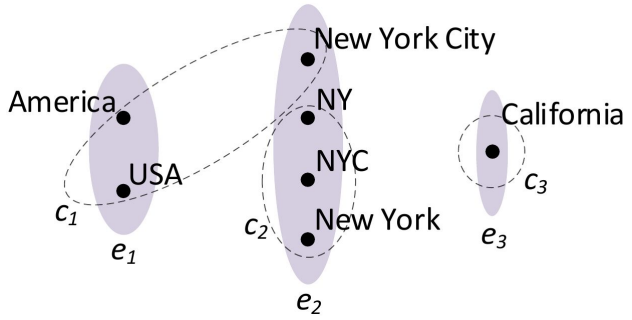
Micro: Micro precision (P_{micro}) is defined as the purity of C clusters [20] based on the assumption that the most frequent gold entity (or relation) in a cluster is correct. Micro recall (R_{micro}) is defined similarly as macro recall.

$$P_{\text{micro}}(C,E) = \frac{1}{N}\sum_{c \in C}\max_{e \in E}|c \cap e|$$
$$R_{\text{micro}}(C,E) = P_{\text{micro}}(E,C)$$

Pairwise: Pairwise precision (P_{pair}) is measured as the ratio of the number of hits in C to the total possible pairs in C. Whereas, pairwise recall (R_{pair}) is the ratio of number of hits in C to all possible pairs in E. A pair of elements in a cluster in C produce a hit if they both refer to the same gold entity (or relation).

$$P_{\text{pair}}(C,E) = \frac{\sum_{c \in C}|\{(v,v') \in e, \exists e \in E, \forall (v,v') \in c\}|}{\sum_{c \in C}{}^{|c|}C_2}$$
$$R_{\text{pair}}(C,E) = \frac{\sum_{c \in C}|\{(v,v') \in e, \exists e \in E, \forall (v,v') \in c\}|}{\sum_{e \in E}{}^{|e|}C_2}$$

Let us illustrate these metrics through a concrete NP canonicalization example shown in Figure 2. In this Figure, we can see that only c_2 and c_3 clusters in C are pure because they contain mentions of only one entity, and hence, $P_{\text{macro}} = \frac{2}{3}$. On the other hand, we have e_1 and e_3 as pure clusters if we interchange the roles of E and C. So, $R_{\text{macro}} = \frac{2}{3}$ in this case. For micro precision, we can see that *America*, *New York*, and *California* are the most frequent gold entities in C clusters. Hence, $P_{\text{micro}} = \frac{6}{7}$. Similarly, $R_{\text{micro}} = \frac{6}{7}$ in this case. For pairwise analysis, we need to first calculate the number of hits in C. In c_1 we have 3 possible pairs out of which only 1, (*America, USA*) is a hit as they belong to same gold cluster e_1. Similarly, we have 3 hits in c_2 and 0 hits in c_3. Hence, $P_{\text{pair}} = \frac{4}{6}$. To compute R_{pair}, we need total number of pairwise decisions in E,



	Precision	Recall	F1
Macro	$\frac{2}{3}$	$\frac{2}{3}$	66.6
Micro	$\frac{6}{7}$	$\frac{6}{7}$	85.7
Pairwise	$\frac{4}{6}$	$\frac{4}{7}$	61.5

Figure 2: Top: Illustrative example for different evaluation metrics. e_i denotes actual clusters, whereas c_i denotes predicted clusters. Bottom: Metric results for the above example. Please see Section 7.2 for details.

which is 1 + 6 + 0 , thus, $R_{\text{pair}} = \frac{4}{7}$. All the results are summarized in Table 2.

For evaluating NP canonicalization, we use Macro, Micro and Pairwise F1 score. However, in the case of relations, where gold labels are not available, we use macro, micro and pairwise precision values based on the scores given by human judges.

7.3 Methods Compared

7.3.1 **Noun Phrase Canonicalization.** For NP canonicalization, CESI has been compared against the following methods:

- **Morphological Normalization:** As used in [12], this involves applying simple normalization operations like removing tense, pluralization, capitalization etc. over NPs and relation phrases.
- **Paraphrase Database (PPDB):** Using PPDB 2.0 [29], we clustered two NPs together if they happened to share a common paraphrase. NPs which could not be found in PPDB are put into singleton clusters.
- **Entity Linking**: Since the problem of NP canonicalization is closely related to entity linking, we compare our method against Stanford CoreNLP Entity Linker [35]. Two NPs linked to the same entity are clustered together.
- **Galárraga-IDF [14]**: IDF Token Overlap was the best performing method proposed in [14] for NP canonicalization. In this method, IDF token similarity is defined between two NPs as in Section 4.1, and HAC is used to cluster the mentions.
- **Galárraga-StrSim [14]**: This method is similar to Galarraga-IDF, but with similarity metric being the Jaro-Winkler [41] string similarity measure.
- **Galárraga-Attr [14]**: Again, this method is similar to the Galarraga-IDF, except that Attribute Overlap is used as the similarity metric between two NPs in this case. Attribute for a NP n, is defined as the set of relation-NP pairs which co-occur with n in the input triples. Attribute overlap similarity between two NPs, is defined as the Jaccard coefficient of the set of attributes:

$$f_{\text{attr}}(n, n') = \frac{|A \cap A'|}{|A \cup A'|}$$

where, A and A' denote the set of attributes associated with n and n'.
Since canonicalization methods using above similarity measures were found to be most effective in [14], even outperforming Machine Learning-based alternatives, we consider these three baselines as representatives of state-of-the-art in Open KB canonicalization.
- **GloVe**: In this scheme, each NP and relation phrase is represented by a 300 dimensional GloVe embedding [30] trained on Wikipedia 2014 and Gigaword 5 [28] datasets with 400k vocabulary size. Word vectors were averaged together to get embeddings for multi-word phrases. These GloVE embeddings were then clustered for final canonicalization.
- **HolE**: In this method, embeddings of NPs and relation phrases in an Open KB are obtained by applying HolE [25] over the Open KB. These embeddings are then clustered to obtain the final canonicalized groupings. Based on the initialization of embeddings, we differentiate between **HolE(Random)** and **HolE(GloVe)**.
- **CESI**: This is the method proposed in this paper, please see Section 3 for more details.

Hyper-parameters: Following [14], we used Hierarchical Agglomerative Clustering (HAC) as the default clustering method across all methods (wherever necessary). For all methods, grid search over the hyperparameter space was performed, and results for the best performing setting are reported. This process was repeated for each dataset.

7.3.2 **Relation Phrase Canonicalization.** AMIE [15] was found to be effective for relation phrase canonicalization in [14]. We thus consider AMIE[1] as the state-of-the-art baseline for relation phrase canonicalization and compare against CESI. We note that AMIE requires NPs of the input Open KB to be already canonicalized. In all our evaluation datasets, we already have *gold* NP canonicalization available. We provide this gold NP canonicalization information as input to AMIE. Please note that CESI doesn't require such pre-canonicalized NP as input, as it performs *joint* NP and relation phrase canonicalization. Moreover, providing gold NP canonicalization information to AMIE puts CESI at a disadvantage. We decided to pursue this choice anyways in the interest of stricter evaluation. However, in spite of starting from this disadvantageous position, CESI significantly outperforms AMIE in relation phrase canonicalization, as we will see in Section 8.1.2.

For evaluating performance of both algorithms, we randomly sampled 25 non-singleton relation clusters for each of the three datasets and gave them to five different human evaluators[2] for assigning scores to each cluster. The setting was kept blind, i.e.,

[1] We use support and confidence values of 2 and 0.2 for all the experiments in this paper.
[2] Authors did not participate in this evaluation.

Method	Base Dataset			Ambiguous Dataset			ReVerb45K			
	Macro	Micro	Pair.	Macro	Micro	Pair.	Macro	Micro	Pair.	Row Average
Morph Norm	58.3	88.3	83.5	49.1	57.2	70.9	1.4	77.7	75.1	62.3
PPDB	42.4	46.9	32.2	37.3	60.2	69.3	46.0	45.4	64.2	49.3
EntLinker	54.9	65.1	75.2	49.7	83.2	68.8	62.8	81.8	80.4	69.1
Galárraga-StrSim	88.2	96.5	97.7	66.6	85.3	82.2	69.9	51.7	0.5	70.9
Galárraga-IDF	94.8	97.9	98.3	67.9	82.9	79.3	71.6	50.8	0.5	71.5
Galárraga-Attr	76.1	51.4	18.1	**82.9**	27.7	8.4	**75.1**	20.1	0.2	40.0
GloVe	95.7	97.2	91.1	65.9	89.9	90.1	56.5	82.9	75.3	82.7
HolE (Random)	69.5	91.3	86.6	53.3	85.0	75.1	5.4	74.6	50.9	65.7
HolE (GloVe)	75.2	93.6	89.3	53.9	85.4	76.7	33.5	75.8	51.0	70.4
CESI	**98.2**	**99.8**	**99.9**	66.2	**92.4**	**91.9**	62.7	**84.4**	**81.9**	**86.3**

Table 2: NP Canonicalization Results. CESI outperforms all other methods across datasets (Best in 7 out of 9 cases. Section 8.1.1)

identity of the algorithm producing a cluster was not known to the evaluators. Based on the average of evaluation scores, precision values were calculated. Only non-singleton clusters were sampled, as singleton clusters will always give a precision of one.

8 RESULTS

In this section, we evaluate the following questions.

Q1. Is CESI effective in Open KB canonicalization? (Section 8.1)
Q2. What is the effect of side information in CESI's performance? (Section 8.2)
Q3. Does addition of entity linking side information degrade CESI's ability to canonicalize unlinked NPs (i.e., NPs missed by the entity linker)? (Section 8.3)

Finally, in Section 8.4, we present qualitative examples and discussions.

8.1 Evaluating Effectiveness of CESI in Open KB Canonicalization

8.1.1 **Noun Phrase Canonicalization.** Results for NP canonicalization are summarized in Table 2. Overall, we find that CESI performs well consistently across the datasets. Morphological Normalization failed to give competitive performance in presence of homonymy. PPDB, in spite of being a vast reservoir of paraphrases, lacks information about real-world entities like people, places etc. Therefore, its performance remained weak throughout all datasets. Entity linking methods make use of contextual information from source text of each triple to link a NP to a KB entity. But their performance is limited because they are restricted by the entities in KB. String similarity also gave decent performance in most cases but since they solely rely on surface form of NPs, they are bound to fail with NPs having dissimilar mentions.

Methods such as Galárraga-IDF, Galárraga-StrSim, and Galárraga-Attr performed poorly on ReVerb45K. Although, their performance is considerably better on the other two datasets. This is because of the fact that in contrast to Base and Ambiguous datasets, ReVerb45K has considerably large number of entities and comparatively fewer triples (Table 1). Galárraga-IDF token overlap is more likely to put two NPs together if they share an uncommon token, i.e., one with high IDF value. Hence, accuracy of the method relies heavily on the quality of document frequency estimates which may be quite misleading when we have smaller number of triples. Similar is the case with Galárraga-Attr which decides similarity of NPs based on the set of shared attributes. Since, attributes for a NP is defined as a set of relation-NP pairs occurring with it across all triples, sparse data also results in poor performance for this method.

GloVe captures semantics of NPs and unlike string similarity it doesn't rely on the surface form of NPs. Therefore, its performance has been substantial across all the datasets. HolE captures structural information from the given triples and uses it for learning embeddings. Through our experiments, we can see that solely structural information from KB is quite effective for NP canonicalization. CESI performs the best across the datasets in 7 out of the 9 settings, as it incorporates the strength of all the listed methods. The superior performance of CESI compared to HolE clearly indicates that the side information is indeed helpful for canonicalization task. Results of GloVe, HolE and CESI suggest that embeddings based method are much more effective for Open KB canonicalization.

	Macro Precision	Micro Precision	Pairwise Precision	Induced Relation Clusters
Base Dataset				
AMIE	42.8	63.6	43.0	7
CESI	**88.0**	**93.1**	**88.1**	**210**
Ambiguous Dataset				
AMIE	55.8	64.6	23.4	46
CESI	**76.0**	**91.9**	**80.9**	**952**
ReVerb45K				
AMIE	69.3	84.2	66.2	51
CESI	**77.3**	**87.8**	**72.6**	**2116**

Table 3: Relation canonicalization results. Compared to AMIE, CESI canonicalizes more number of relation phrases at higher precision. Please see Section 8.1.2 for details.

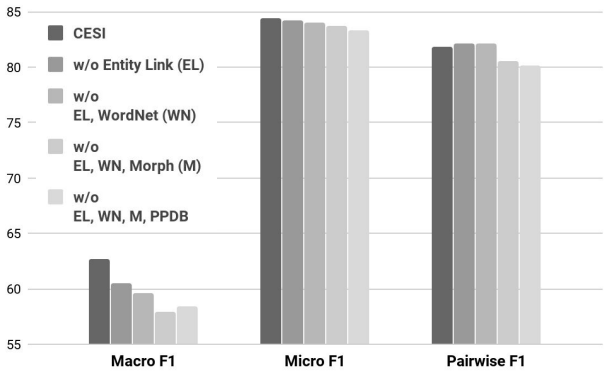


Figure 3: Performance comparison of various side information-ablated versions of CESI for NP canonicalization in the ReVerb45K dataset. Overall, side information helps CESI improve performance. Please see Section 8.2 for details.

8.1.2 **Relation Phrase Canonicalization.** Results for relation phrase canonicalization are presented in Table 3. For all experiments, in spite of using quite low values for minimum support and confidence, AMIE was unable to induce any reasonable number of non-singleton clusters (e.g., only 51 clusters out of the 22K relation phrases in the ReVerb45K dataset). For relation canonicalization experiments, AMIE was evaluated on gold NP canonicalized data as the algorithm requires NPs to be already canonicalized. CESI, on the other hand, was tested on all the datasets without making use of gold NP canonicalization information.

Based on the results in Table 3, it is quite evident that AMIE induces too few relation clusters to be of value in practical settings. On the other hand, CESI consistently performs well across all the datasets and induces significantly larger number of clusters.

8.2 Effect of Side Information in CESI

In this section, we evaluate the effect of various side information in CESI's performance. For this, we evaluated the performances of various versions of CESI, each one of them obtained by ablating increasing amounts of side information from the full CESI model. Experimental results comparing these ablated versions on the ReVerb45K are presented in Figure 3. From this figure, we observe that while macro performance benefits most from different forms of side information, micro and pairwise performance also show increased performance in the presence of various side information. This validates one of the central thesis of this paper: side information, along with embeddings, can result in improved Open KB canonicalization.

8.3 Effect of Entity Linking Side Information on Unlinked NP Canonicalization

From experiments in Section 8.2, we find that Entity Linking (EL) side information (see Section 4.1) is one of the most useful side information that CESI exploits. However, such side information is not available in case of unlinked NPs, i.e., NPs which were not linked by the entity linker. So, this naturally raises the following question: does CESI overfit to the EL side information and ignore the unlinked NPs, thereby resulting in poor canonicalization of such unlinked NPs?

In order to evaluate this question, we compared CESI's performance on unlinked NPs in the ReVerb45K dataset, with and without EL side information. We note that triples involving unlinked NPs constitute about 25% of the entire dataset. Results are presented in Table 4. From this table, we observe that CESI doesn't overfit to EL side information, and it selectively uses such information when appropriate (i.e., for linked NPs). Because of this robust nature, presence of EL side information in CESI doesn't have an adverse effect on the unlinked NPs, in fact there is a small gain in performance.

	Macro F1	Micro F1	Pairwise F1
CESI	81.7	87.6	81.5
CESI w/o EL	81.3	87.3	80.7

Table 4: CESI's performance in canonicalizing unlinked NPs, with and without Entity Linking (EL) side information, in the ReVerb45K dataset. We observe that CESI does not overfit to EL side information, and thereby helps prevent performance degradation in unlinked NP canonicalization (in fact it even helps a little). Please see Section 8.3 for details.

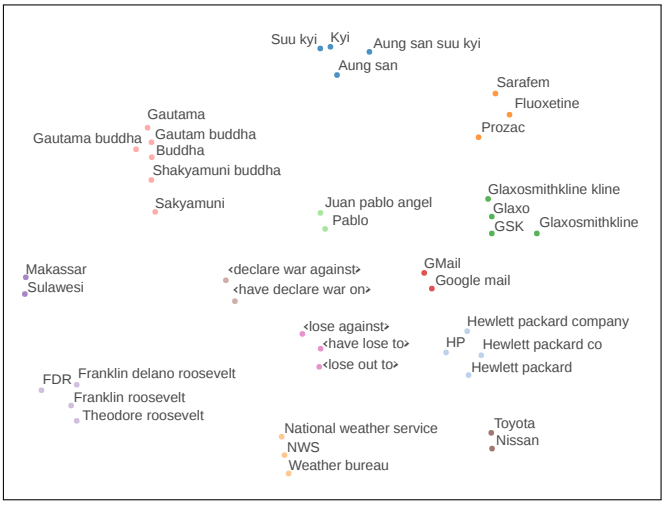


Figure 4: t-SNE visualization of NP and relation phrase (marked in '< ··· >') embeddings learned by CESI for ReVerb45K dataset. We observe that CESI is able to induce non-trivial canonical clusters. Please see Section 8.4 for details.

8.4 Qualitative Evaluation

Figure 4 shows some of the NP and relation phrase clusters detected by CESI in ReVerb45K dataset. These results highlight the efficacy of algorithm in canonicalizing non-trivial NPs and relation phrases. The figure shows t-SNE [40] visualization of NP and relation phrase (marked in '< ··· >') embeddings for a few examples. We can see

that the learned embeddings are actually able to capture equivalence of NPs and relation phrases. The algorithm is able to correctly embed *Prozac*, *Sarafem* and *Fluoxetine* together (different names of the same drug), despite their having completely different surface forms.

Figure 4 also highlights the failures of CESI. For example, *Toyota* and *Nissan* have been embedded together although the two being different companies. Another case is with *Pablo* and *Juan Pablo Angel*, which refer to different entities. The latter case can be avoided by keeping track of the source domain type information of each NP for disambiguation. In this if we know that *Juan Pablo Angel* has come from *SPORTS* domain, whereas *Pablo* has come from a different domain then we can avoid putting them together. We tried using DMOZ [34] dataset, which provide mapping from URL domain to their categories, for handling such errors. But, because of poor coverage of URLs in DMOZ dataset, we couldn't get significant improvement in canonicalization results. We leave this as a future work.

9 CONCLUSION

Canonicalizing Open Knowledge Bases (KBs) is an important but underexplored problem. In this paper, we proposed CESI, a novel method for canonicalizing Open KBs using learned embeddings and side information. CESI solves a joint objective to learn noun and relation phrase embeddings, while utilizing relevant side information in a principled manner. These learned embeddings are then clustered together to obtain canonicalized noun and relation phrase clusters. In this paper, we also propose ReVerb45K, a new and larger dataset for Open KB canonicalization. Through extensive experiments on this and other real-world datasets, we demonstrate CESI's effectiveness over state-of-the-art baselines. CESI's source code and all data used in the paper are publicly available[3].

ACKNOWLEDGEMENT

We thank the reviewers for their constructive comments. This work is supported in part by MHRD, Govt. of India, and by gifts from Google Research and Accenture. We thank Anand Mishra and other members of MALL Lab, IISc for carefully reading drafts of this paper.

REFERENCES

[1] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. DBpedia: A Nucleus for a Web of Open Data. In *Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference (ISWC'07/ASWC'07)*. Springer-Verlag, Berlin, Heidelberg, 722–735. http://dl.acm.org/citation.cfm?id=1785162.1785216

[2] Satanjeev Banerjee and Ted Pedersen. 2002. *An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet.* Springer Berlin Heidelberg, Berlin, Heidelberg, 136–145. https://doi.org/10.1007/3-540-45715-1_11

[3] Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open Information Extraction from the Web. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence.*

[4] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data (SIGMOD '08)*. ACM, New York, NY, USA, 1247–1250. https://doi.org/10.1145/1376616.1376746

[5] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 2787–2795. http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf

[6] Matthias Bröcheler, Lilyana Mihalkova, and Lise Getoor. 2010. Probabilistic Similarity Logic. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence (UAI'10)*. AUAI Press, Arlington, Virginia, United States, 73–82. http://dl.acm.org/citation.cfm?id=3023549.3023558

[7] Jamie Callan, Mark Hoy, Changkuk Yoo, and Le Zhao. 2009. Clueweb09 data set. (2009).

[8] Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2011. An Analysis of Open Information Extraction Based on Semantic Role Labeling. In *Proceedings of the Sixth International Conference on Knowledge Capture (K-CAP '11)*. ACM, New York, NY, USA, 113–120. https://doi.org/10.1145/1999676.1999697

[9] Daniel Defays. 1977. An efficient algorithm for a complete link method. *Comput. J.* 20, 4 (1977), 364–366.

[10] Claudio Delli Bovi, Luis Espinosa Anke, and Roberto Navigli. 2015. Knowledge Base Unification via Sense Embeddings and Disambiguation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, 726–736. http://aclweb.org/anthology/D15-1084

[11] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. 2014. Knowledge Vault: A Web-scale Approach to Probabilistic Knowledge Fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '14)*. ACM, New York, NY, USA, 601–610. https://doi.org/10.1145/2623330.2623623

[12] Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying Relations for Open Information Extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP '11)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1535–1545. http://dl.acm.org/citation.cfm?id=2145432.2145596

[13] Evgeniy Gabrilovich, Michael Ringgaard, and Amarnag Subramanya. 2013. FACC1: Freebase annotation of ClueWeb corpora, Version 1. *Release date* (2013), 06–26.

[14] Luis Galárraga, Geremy Heitz, Kevin Murphy, and Fabian M. Suchanek. 2014. Canonicalizing Open Knowledge Bases. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management (CIKM '14)*. ACM, New York, NY, USA, 1679–1688. https://doi.org/10.1145/2661829.2662073

[15] Luis Antonio Galárraga, Christina Teflioudi, Katja Hose, and Fabian Suchanek. 2013. AMIE: Association Rule Mining Under Incomplete Evidence in Ontological Knowledge Bases. In *Proceedings of the 22Nd International Conference on World Wide Web (WWW '13)*. ACM, New York, NY, USA, 413–422. https://doi.org/10.1145/2488388.2488425

[16] Ari Kobren, Nicholas Monath, Akshay Krishnamurthy, and Andrew McCallum. 2017. A Hierarchical Algorithm for Extreme Clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17)*. ACM, New York, NY, USA, 255–264. https://doi.org/10.1145/3097983.3098079

[17] Jayant Krishnamurthy and Tom M. Mitchell. 2011. Which Noun Phrases Denote Which Concepts?. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT '11)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 570–580. http://dl.acm.org/citation.cfm?id=2002472.2002545

[18] Dekang Lin and Patrick Pantel. 2001. DIRT @SBT@Discovery of Inference Rules from Text. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '01)*. ACM, New York, NY, USA, 323–328. https://doi.org/10.1145/502512.502559

[19] Thomas Lin, Mausam, and Oren Etzioni. 2012. Entity Linking at Web Scale. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX '12)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 84–88. http://dl.acm.org/citation.cfm?id=2391200.2391216

[20] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval.* Cambridge University Press, New York, NY, USA.

[21] Mausam Mausam. 2016. Open Information Extraction Systems and Downstream Applications. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI'16)*. AAAI Press, 4074–4077. http://dl.acm.org/citation.cfm?id=3061053.3061220

[22] George A. Miller. 1995. WordNet: A Lexical Database for English. *Commun. ACM* 38, 11 (Nov. 1995), 39–41. https://doi.org/10.1145/219717.219748

[23] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling. 2015. Never-Ending Learning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*.

[3] https://github.com/malllabiisc/cesi

[24] Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. PATTY: A Taxonomy of Relational Patterns with Semantic Types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL '12)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1135–1145. http://dl.acm.org/citation.cfm?id=2390948.2391076

[25] Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. 2016. Holographic Embeddings of Knowledge Graphs. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI'16)*. AAAI Press, 1955–1961. http://dl.acm.org/citation.cfm?id=3016100.3016172

[26] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A Three-way Model for Collective Learning on Multi-relational Data. In *Proceedings of the 28th International Conference on International Conference on Machine Learning (ICML'11)*. Omnipress, USA, 809–816. http://dl.acm.org/citation.cfm?id=3104482.3104584

[27] Madhav Nimishakavi, Uday Singh Saini, and Partha Talukdar. 2016. Relation Schema Induction using Tensor Factorization with Side Information. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 414–423. https://doi.org/10.18653/v1/D16-1040

[28] Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. *English gigaword fifth edition, linguistic data consortium.* Technical Report. Technical report, Technical Report. Linguistic Data Consortium, Philadelphia.

[29] Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*. 425–430. http://aclweb.org/anthology/P/P15/P15-2070.pdf

[30] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. 1532–1543. http://www.aclweb.org/anthology/D14-1162

[31] Jay Pujara, Hui Miao, Lise Getoor, and William Cohen. 2013. Knowledge Graph Identification. In *Proceedings of the 12th International Semantic Web Conference - Part I (ISWC '13)*. Springer-Verlag New York, Inc., New York, NY, USA, 542–557. https://doi.org/10.1007/978-3-642-41335-3_34

[32] Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and Global Algorithms for Disambiguation to Wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1 (HLT '11)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1375–1384. http://dl.acm.org/citation.cfm?id=2002472.2002642

[33] Swarnadeep Saha, Harinder Pal, and Mausam. 2017. Bootstrapping for Numerical Open IE. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, 317–323. https://doi.org/10.18653/v1/P17-2050

[34] Gaurav Sood. 2016. Parsed DMOZ data. (2016). https://doi.org/10.7910/DVN/OMV93V

[35] Valentin I. Spitkovsky and Angel X. Chang. 2012. A Cross-Lingual Dictionary for English Wikipedia Concepts. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)* (23-25), Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet UÄ§ur DoÄ§an, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis (Eds.). European Language Resources Association (ELRA), Istanbul, Turkey.

[36] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A Core of Semantic Knowledge. In *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*. ACM, New York, NY, USA, 697–706. https://doi.org/10.1145/1242572.1242667

[37] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance Multi-label Learning for Relation Extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL '12)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 455–465. http://dl.acm.org/citation.cfm?id=2390948.2391003

[38] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. 2005. *Introduction to Data Mining, (First Edition).* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[39] Salvatore Trani, Diego Ceccarelli, Claudio Lucchese, Salvatore Orlando, and Raffaele Perego. 2014. Dexter 2.0: An Open Source Tool for Semantically Enriching Data. In *Proceedings of the 2014 International Conference on Posters & Demonstrations Track - Volume 1272 (ISWC-PD'14)*. CEUR-WS.org, Aachen, Germany, Germany, 417–420. http://dl.acm.org/citation.cfm?id=2878453.2878558

[40] L.J.P. van der Maaten and G.E. Hinton. 2008. Visualizing High-Dimensional Data Using t-SNE. (2008).

[41] William E Winkler. 1999. The state of record linkage and current research problems. In *Statistical Research Division, US Census Bureau*. Citeseer.

[42] Alexander Yates and Oren Etzioni. 2009. Unsupervised Methods for Determining Object and Relation Synonyms on the Web. *J. Artif. Int. Res.* 34, 1 (March 2009), 255–296. http://dl.acm.org/citation.cfm?id=1622716.1622724