Improving Knowledge Base Construction from Robust Infobox Extraction

Boya Peng* **Yejin Huh** **Xiao Ling** **Michele Banko***

Apple Inc.
1 Apple Park Way
Cupertino, CA, USA
{yejin.huh,xiaoling}@apple.com

Sentropy Technologies
{emma,mbanko}@sentropy.io*

Abstract

A capable, automatic Question Answering (QA) system can provide more complete and accurate answers using a comprehensive knowledge base (KB). One important approach to constructing a comprehensive knowledge base is to extract information from Wikipedia infobox tables to populate an existing KB. Despite previous successes in the Infobox Extraction (IBE) problem (*e.g.*, DBpedia), three major challenges remain: 1) Deterministic extraction patterns used in DBpedia are vulnerable to template changes; 2) Over-trusting Wikipedia anchor links can lead to entity disambiguation errors; 3) Heuristic-based extraction of unlinkable entities yields low precision, hurting both accuracy and completeness of the final KB. This paper presents a robust approach that tackles all three challenges. We build probabilistic models to predict relations between entity mentions directly from the infobox tables in HTML. The entity mentions are linked to identifiers in an existing KB if possible. The unlinkable ones are also parsed and preserved in the final output. Training data for both the relation extraction and the entity linking models are automatically generated using distant supervision. We demonstrate the empirical effectiveness of the proposed method in both precision and recall compared to a strong IBE baseline, DBpedia, with an absolute improvement of 41.3% in average F_1. We also show that our extraction makes the final KB significantly more complete, improving the completeness score of list-value relation types by 61.4%.

1 Introduction

Most existing knowledge bases (KBs) are largely incomplete. This can be seen in Wikidata (Vrandečić and Krötzsch, 2014), which is a widely used knowledge graph created largely by human editors. Only 46% of person entities in Wikidata have birth places available [1]. An estimate of 584 million facts are maintained in Wikipedia, not in Wikidata (Hellmann, 2018). A downstream application such as Question Answering (QA) will suffer from this incompleteness, and fail to answer certain questions or even provide an incorrect answer especially for a question about a list of entities due to a closed-world assumption. Previous work on enriching and growing existing knowledge bases includes relation extraction on natural language text (Wu and Weld, 2007; Mintz et al., 2009; Hoffmann et al., 2011; Surdeanu et al., 2012; Koch et al., 2014), knowledge base reasoning from existing facts (Lao et al., 2011; Guu et al., 2015; Das et al., 2017), and many others (Dong et al., 2014).

Wikipedia (https://wikipedia.org) has been one of the key resources used for knowledge base construction. In many Wikipedia pages, a summary table of the subject, called an infobox table, is presented in the top right region of the page (see the leftmost table in Figure 1 for the infobox table of The_Beatles). Infobox tables offer a unique opportunity for extracting information and populating a knowledge base. An infobox table is structurally formatted as an HTML table and therefore it is often not necessary to parse the text into a syntactic parse tree as in natural language extraction. Intra-Wikipedia anchor links are prevalent in infobox tables, often providing unambiguous references to entities. Most importantly, a significant amount of information represented in the infobox tables are not otherwise available in a more traditional structured knowledge base, such as Wikidata.

We are not the first to use infobox tables

[*] Work performed at Apple Inc.

[1] as of June 2018

for knowledge base completion. The pioneering work of DBpedia (Auer et al., 2007; Lehmann et al., 2015)[2] extracts canonical knowledge triples (`subject`, *relation type*, `object`) from infobox tables with a manually created mapping from Mediawiki [3] templates to relation types. Despite the success of the DBpedia project, three major challenges remain. First, deterministic mappings are sensitive to template changes. If Wikipedia modifies an infobox template (*e.g.*, the attribute "birthDate" is renamed to "dateOfBirth"), the DBpedia mappings need to be manually updated. Secondly, while Wikipedia anchor links facilitate disambiguation of string values in the infobox tables, blindly trusting the anchor links can cause errors. For instance, both "Sasha" and "Malia," children of `Barack Obama`, are linked to a section of the Wikipedia page of `Barack_Obama`, rather than their own pages. Finally, little attention has been paid to the extraction of unlinkable entities. For example, `Larry King` has married seven women, only one of which can be linked to a Wikipedia page. A knowledge base without the information of the other six entities will provide an incorrect answer to the question "How many women has Larry King married?"

In this paper, we present a system, RIBE, to tackle all three challenges: 1) We build probabilistic models to predict relations and object entity links. The learned models are more robust to changes in the underlying data representation than manually maintained mappings. 2) We incorporate the information from HTML anchors and build an entity linking system to link string values to entities rather than fully relying on the anchor links. 3) We produce high-quality extractions even when the objects are unlinkable, which improves the completeness of the final knowledge base.

We demonstrate that the proposed method is effective in extracting over 50 relation types. Compared to a strong IBE baseline, DBpedia, our extractions achieve significant improvements on both precision and recall, with a 41.3% increase in average F_1 score. We also show that the extracted triples add a great value to an existing knowledge base, Wikidata, improving an average recall of list-value relation types by 61.4%.

To summarize, our contributions are three-fold:

- RIBE produces high-quality extractions, achieving higher precision and recall compared to DBpedia.
- Our extractions make Wikidata more complete by adding a significant number of triples for 51 relation types.
- RIBE extracts relations with unlinkable entities, which are crucial for the completeness of list-value relation types and the question answering capability from a knowledge base.

2 Related Work

Auer and Lehmann (2007) proposed to extract information from infobox tables by pattern matching against Mediawiki templates and parsing and transforming them into RDF triples. However, relation types of the triples remain lexical and can be ambiguous. Lehmann et al. (2015) introduced an ontology to reduce the ambiguity in relation types. A mapping from infobox attributes to the ontology [4] is manually created, which is brittle to template changes. In contrast, RIBE is much more robust. It trains statistical models with distant supervision from Wikidata to automatically learn the mapping from infobox attributes to Wikidata relation types. RIBE properly parses infobox values into separate object mentions, instead of relying on existing Mediawiki boundaries as in (Auer and Lehmann, 2007; Lehmann et al., 2015). The RIBE entity linker learns to make entity link predictions rather than a direct translation of anchor links (Auer and Lehmann, 2007; Lehmann et al., 2015) vulnerable to human edit errors. While there is other relevant work, due to space constraints, it is discussed throughout the paper and in the appendices as appropriate.

3 Problem Definition

We define a **relation** as a triple (e_1, r, e_2) or $r(e_1, e_2)$, where r is the **relation type** and e_1, e_2 are the **subject** and the **object** entities respectively [5]. We define an **entity mention** m as the surface form of an entity, and a **relation mention**

[2] Throughout the paper, we use "DBpedia" to refer to its infobox extraction component rather than the DBpedia knowledge base unless specified. We use Wikidata as the baseline knowledge base for our experiments since it is updated more frequently. The last release DBpedia knowledge base was in 2016.

[3] Mediawiki is a markup language that defines the page layout and presentation of Wikipedia pages.

[4] DBpedia Mappings Wiki at `http://mappings.dbpedia.org`.

[5] e_2 can also be a literal value such as a number or a time expression unless specified.

as a pair of entity mentions (m_1, m_2) for a relation type r. We denote the set of infobox tables by $T = \{t_1, t_2, ..., t_n\}$ where t_i appears in the Wikipedia page of the entity e_i [6]. The Infobox Extraction problem studied in this paper aims at extracting a set of relation triples $r(e_1, e_2)$ from an input of Wikipedia infobox tables T.

4 System Overview

We describe the RIBE system in this section. Wikidata (Vrandečić and Krötzsch, 2014) is employed as the base external KB. We draw distant supervision from it by matching candidate mentions against Wikidata relation triples for training statistical models. We also link our mentions to Wikidata entities [7] and compare our extractions to it for evaluation (Tables 4, 8). The final output of RIBE is a set of relation triples $R = \{r_i(e_1^i, e_2^i)\}$.

4.1 Relation Extraction

Figure 1 depicts an overview of the relation extractor. It extracts relation mentions $r(m_1, m_2)$ from each infobox table in T in four stages: entity mention generation, feature generation, distant supervision, and model training and inference.

4.1.1 Mention Generation

We parse each infobox table rendered in HTML, instead of the raw Mediawiki template, to generate object mentions. As the infobox templates evolve over time, different Mediawiki templates have been used to describe the same type of things. Despite the difference in templates, the rendered infobox tables in HTML displays a surprisingly consistent format: each row contains an attribute cell and an attribute value cell.

We chunk the text of each attribute value cell and generate object mentions using a few heuristics such as HTML tag boundaries. Table 5 in Appendix A shows the heuristics. Each pair of subject and object mentions becomes one relation mention candidate. Note that an off-the-shelf noun phrase chunker (Sang and Buchholz, 2000) or a Named Entity Recognition (NER) tagger (Finkel et al., 2005) doesn't work well in this case as they are often trained on grammatical sentences instead of the textual segments seen in infobox tables.

4.1.2 Feature Generation

For each relation mention, we generate features similar to the ones from Mintz et al. (2009), with some modifications. Since the subject of a relation mention is outside the infobox, most generated features focus on the object (*e.g.*, the word shape of the object mention), and its context (*e.g.*, the infobox attribute value). Table 6 in Appendix A lists the complete set of features.

4.1.3 Distant Supervision

Instead of manually collecting training examples, we use distant supervision (Craven and Kumlien, 1999; Bunescu and Mooney, 2007; Mintz et al., 2009) from Wikidata to automatically generate training data for relation extraction. We assume that a pair of mentions (m_{e_1}, m_2) expresses the relation type r if we are able to match m_2 to e_2 where $r(e_1, e_2)$ is a Wikidata relation. Since the object mention m_2 is non-canonical, we do a string match between the entity name and the mention text or a direct entity match if an anchor link exists for the mention. We construct the negative training data for a relation type r by combining positive examples of other relation types and a random sample of the unlabeled entity mention pairs.

4.1.4 Model Training and Inference

We train a binary logistic regression classifier (Friedman et al., 2001) for each relation type. The classifier predicts how likely a pair of entity mentions is to express a relation type r. We only output relations with probabilities higher than a threshold θ (0.9 is used empirically). Otherwise, a mention pair is deemed to not express any relation type. We choose one binary classifier for each type rather than a single multi-class classifier because one pair of mentions can express multiple relation types. For example, the mention pair ("The Beatles", "England") under the attribute "Origin" (see Fig. 1) expresses two relation types: *country of origin* and *location of formation*.

4.2 Entity Linking & Normalization

Figure 2 provides an overview of the entity linker in four stages: candidate generation, feature generation, distant supervision, and model training and inference. The subject m_{e_i} of each extracted relation mention $r(m_{e_i}, m_2)$ is trivially linked to the subject entity e_i. The entity linker links the object mention to a Wikidata entity unless no corresponding entity exists, in which case we mark

[6] We assume that only one representative infobox table is available on each page.

[7] Note that each Wikipedia page has a corresponding Wikidata entry. Exceptions exist but are negligible. In this paper, we use "Wikipedia entity" and "Wikidata entity" interchangeably for the same real-world entity.

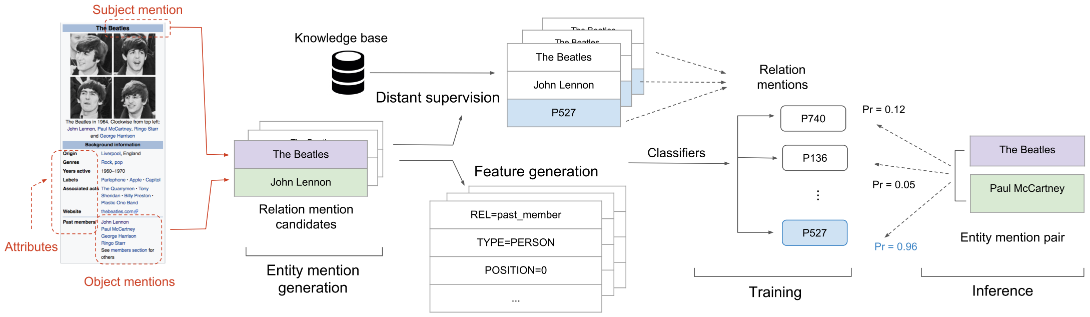

Figure 1: Relation Extractor Overview

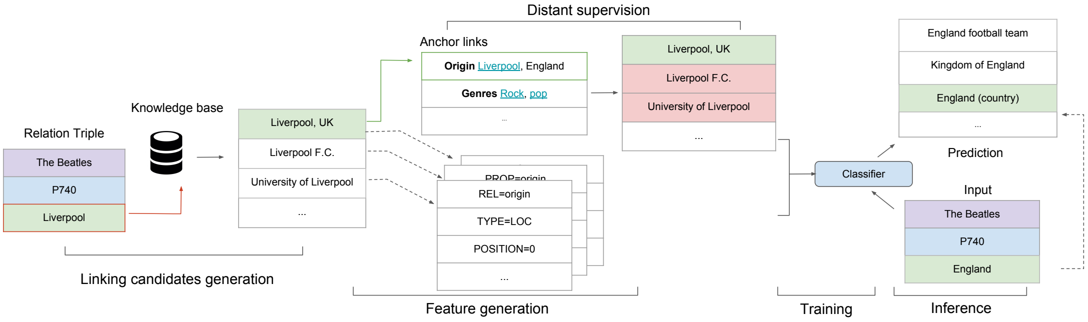

Figure 2: Entity Linker Overview

m_2 an unlinkable entity. We normalize the literal values when m_2 is not an entity reference.[8]

4.2.1 Candidate Generation

We generate candidate entities $\{\tilde{e}_2^i\}$ for m_2 in each $r(m_{e_1}, m_2)$ extracted. Candidate entities are generated from the anchor link associated with the mention if present, matching the surface form text of anchor links harvested from other pages for the same relation type r, and type-aware name matching with Wikidata entities. We determine the entity type(s) of the object for a relation type r from the statistics of existing relation triples in Wikidata. In this work, we use a set of coarse-grained entity types: person, band, school, organization, location, event, creative work, and award.[9]

4.2.2 Feature Generation

We generate features for each candidate entity of m_2, including lexical features of m_2, textual similarity between the entity name and m_2, how the candidate entity is generated (*e.g.*, anchor link or name matching), and for name matching, the kind of names the mention is matched against (*e.g.*, harvested surface form text or an entity alias). Contextual information surrounding m_2 in the same infobox is also used for disambiguation.[10]

4.2.3 Distant Supervision

Similar to Sec. 4.1.3, we use Wikidata to distantly supervise the entity linker. We compare each $r(m_{e_1}, \tilde{e}_2^i)$ triple with all $r(e_1, e_2)$ triples from Wikidata. The candidate entity in which $\tilde{e}_2^i = e_2$ is labeled as a positive example, while the rest of the candidate entities are considered negative examples. Note that the supervision is noisy because there may be multiple valid or very similar canonical entities for the same $r(m_{e_1}, m_2)$, especially for relation types such as *genre*. Additional heuristic rules are used to denoise the supervision (cf. *labeling functions* in Ratner et al. (2016)).

4.2.4 Model training & Inference

We use a logistic regression model to predict the probability of each candidate for a mention. The candidate entity with the highest probability for a mention is chosen. If the highest probability does not meet a threshold (0.5 is chosen empirically), we mark a mention an unlinkable entity. For location-type objects, we make a collective prediction that the entity choices of neighboring mentions are decided altogether (see Appendix B.2).

[8] For relation types with quantity objects such as *population* and *height*, date objects such as *birth date*, or string values such as *website*, we normalize them into a canonical form with respect to each data type.

[9] When the required entity type is absent from this set (*e.g.*, an object value of the *language spoken* relation type), no candidate is provided from type-aware name matching.

[10] See feature examples in Table 7 of Appendix A.

5 Experiments

In this section, we conduct experiments to answer the following questions:

- Does RIBE produce high-quality extractions? (Sec. 5.2)
- Are these extractions a significant addition to Wikidata? (Sec. 5.3)
- How does the extraction with unlinkable entities affect the quality of the KB especially for the list-valued relation types? (Sec. 5.4)

5.1 Data

We use Wikipedia infobox tables and Wikidata [11] to construct our training and evaluation data set. We test RIBE on both coarse-grained entity types such as *person*, *location* and *event* as well as fine-grained ones such as *band* and *school*. We denote the set of entity types ET. A set of 51 Wikidata relation types RT is used as extraction targets, a sample of which is shown in Table 4 [12].

To assign types to entities, we first recursively traverse Wikidata to collect all sub-types of a target entity type $t \in ET$ via the *subclass of* relation. For example, both *city* and *state* are subclasses of type *location*. Next, we use the *instance of* relation type to identify types for entities. An entity is deemed type t if and only if it appears as the subject of an *instance of* relation, where the object is one of the sub-types of t. Around 1.2 million infobox tables remain after a subject type filtering. Table 1 displays a type breakdown.

Subject entity type	*Percentage (%)*
Person	81.5
Event	10.4
Location	3.3
Band	2.7
School	1.9

Table 1: Infobox subject entity type breakdown.

Subject:Relation type	**Wikidata**	**DBpedia**
per:team	P54	team
per:place_of_birth	P19	birthPlace
per:date_of_birth	P569	birthDate
per:occupation	P106	occupation
per:place_of_death	P20	deathPlace
per:date_of_death	P570	deathDate
per:educated_at	P69	almaMater
band:has_member	P527	bandMember formerBandMember currentMember pastMember
per:spouse	P26	spouse
per:award	P166	award
per:child	P40	child
per:political_party	P102	party
per:partner	P551	residence
per:record_label	P264	recordLabel*
per:parent	K206	parent
band:genre	P136	genre
per:instrument	P1303	instrument
band:record_label	P264	recordLabel*
per:employer	P108	employer
per:burial_place	P119	placeOfBurial
school:student_count	P2196	numberOfStudents
band:country	P17	country
event:country	P17	country
loc:largest_city	K223	largestCity
loc:ethic_group	P172	ethnicGroup*

Table 2: Mappings from Wikidata relation types to DBpedia ones. The relation types followed by * are missing from the output generated by the DBpedia extraction code but present in the latest public DBpedia data release. We use the latter for those relation types for a fair comparison.

5.2 Extraction Quality

We compare RIBE with a strong IBE baseline, DBpedia (Lehmann et al., 2015). We obtain extractions from DBpedia by running their extractor[13] on the same set of 1.2 million infobox tables. We identified 74 mappings of relation types between DBpedia and Wikidata from the DBpedia ontology[14]. 23 of them overlap with our target relation set RT (see Table 3). We divide DBpedia relation types by subject types. For instance, we create two sub-types *person:record_label* and *band:record_label* from the DBpedia relation type *recordLabel*, where the first one includes *recordLabel* relations with subject type *person*, and the second one with *band*. Table 2 lists the complete mappings of relation types from Wikidata to DBpedia.

5.2.1 Evaluation Methodology

We evaluate the extraction quality for each relation type $r \in RT$ using four metrics: yield, precision, recall, and F_1 score. **Yield** (Y) is defined as the number of all uniquely extracted relations $r(e_1, e_2)$. To compute precision and recall, we first collect a ground truth set of relations $G = \{G_r | r \in RT\}$. [15] We create a union of extractions from RIBE and DBpedia, and randomly sample around 100 entities with at least one relation extracted from either system. This is similar to the pooling methodology used in the TAC KBP evaluation process (TAC, 2017). The sampled extractions are graded by human annotators.

[11] An 11/2018 Wikipedia and a 06/2018 Wikidata dump.

[12] The full list is provided in Table 8 in Appendix C

[13] https://github.com/dbpedia/extraction-framework/

[14] https://wiki.dbpedia.org/Downloads2014#h395-1

[15] If both systems have low true recall, this ground truth set will have low recall as well. While we leave a better estimate of recall for future work, anecdotally we see that that is not the case here.

Subject	*Relation Type*	*Yield*		*P / R / F_1 (%)*		*C (%)*	
		RIBE	*DBpedia*	RIBE	*DBpedia*	RIBE	*DBpedia*
person	*team*	1,606,179	27598	97.9 / 99.4 / 98.6	42.8 / 0.8 / 1.5	97.6	1.1
person	*place of birth*	1,222,396	733498	99.3 / 80.5 / 88.9	100.0 / 51.6 / 68.0	74.0	39.4
person	*date of birth*	878,537	478383	100.0 / 98.1 / 99.0	100.0 / 48.6 / 65.4	-	-
person	*occupation*	541,901	355723	99.4 / 98.9 / 99.1	34.4 / 16.5 / 22.3	97.8	13.9
person	*place of death*	445,128	226721	98.7 / 97.5 / 98.1	98.8 / 54.0 / 69.8	95.6	45.0
person	*birth name*	325,594	89574	98.9 / 97.0 / 97.9	100.0 / 25.0 / 40.0	97.0	25.0
person	*date of death*	317,489	187762	100.0 / 98.0 / 98.9	100.0 / 57.6 / 73.0	-	-
person	*educated at*	279,661	100916	93.3 / 96.2 / 94.7	100.0 / 36.9 / 53.9	94.0	32.6
band	*has member*	186,090	49485	97.6 / 98.8 / 98.1	97.4 / 19.2 / 32.0	92.7	11.4
person	*spouse*	156,984	40312	96.6 / 98.3 / 97.4	92.5 / 21.4 / 34.7	97.8	17.3
person	*award*	119,136	91158	91.5 / 92.6 / 92.0	73.6 / 58.3 / 65.0	85.7	52.3
person	*child*	115,582	30441	97.6 / 96.2 / 96.8	98.1 / 25.5 / 40.4	93.4	25.0
person	*political party*	111,061	45528	100.0 / 96.0 / 97.9	100.0 / 36.6 / 53.5	95.5	31.4
person	*partner*	94,541	57354	100.0 / 89.8 / 94.6	100.0 / 56.7 / 72.3	87.7	51.1
person	*record label*	82,478	46440	99.1 / 96.2 / 97.6	100.0 / 52.8 / 69.1	90.7	38.1
person	*parent*	79,497	36105	98.0 / 94.4 / 96.1	93.6 / 36.9 / 52.9	92.6	33.6
band	*genre*	78,742	66843	99.2 / 98.8 / 99.0	99.1 / 76.7 / 86.4	96.9	84.6
person	*instrument*	69,816	42150	100.0 / 94.2 / 97.0	96.1 / 40.9 / 57.3	91.8	36.7
band	*record label*	64,148	37883	98.5 / 97.2 / 97.8	97.6 / 59.2 / 73.7	93.4	46.7
person	*employer*	62,183	8726	93.2 / 96.8 / 94.9	92.5 / 16.1 / 27.4	95.7	20.0
person	*place of burial*	54,430	1141	98.6 / 98.7 / 98.6	100.0 / 4.0 / 7.6	97.8	4.3
school	*student count*	20,410	24624	98.6 / 79.6 / 88.1	98.7 / 83.9 / 90.7	-	-
event	*country*	12,625	9028	100.0 / 89.3 / 94.3	50.0 / 23.8 / 32.2	88.7	23.7
location	*largest city*	914	3205	100.0 / 24.5 / 39.3	100.0 / 89.8 / 94.6	-	-
location	*ethnic group*	753	147	91.1 / 90.8 / 90.9	83.9 / 16.8 / 27.9	84.1	18.8
Average				**97.9 / 91.9 / 93.8**	89.9 / 40.3 / 52.5	**92.4**	31.0

Table 3: Comparison with DBpedia. *P*/*R*/F_1 denotes Precision/Recall/F1 measures. *C* denotes the completeness score for list-value relation types. An asterisk marks the relation types that are missing from the output generated by the DBpedia extraction code but present in the last public DBpedia release at `https://wiki.dbpedia.org/develop/datasets/dbpedia-version-2016-10`. We use the latter for those relation types to conduct a fair comparison.

According to our annotation guidelines, we mark an extraction incorrect if one of the the following is met.

- The relation is not expressed in the infobox.
- The object entity has an incorrect identifier.
- The object of a relation triple is unlinked by the system but should be linked in the ground truth. For instance, a string "United States" not linked to its entity identifier in Wikidata is considered incorrect.
- The object is incorrectly parsed. For example, "Sasha and Malia" would be an incorrect extraction for the *child* relation of `Barack Obama`.

The final set G_r consists of all correct relations of the sampled entities from both approaches. The number of labels varies from 83 for *event:country* to 557 for *band:has_member*, resulting in a total of 4,858 labels on triples and an average of 194 per relation type. The **Precision** (P) of a system s, RIBE or DBpedia, is computed as $P_r^s = \frac{|U_r^s \cap G_r|}{|U_r^s|}$, where U_r^s is the set of all extracted relations of r by system s. An absolute **Recall** (R) of the universe is difficult to compute. We compute an estimated recall $R_r^s = \frac{|U_r^s \cap G_r|}{|G_r|}$ w.r.t the ground truth set. The standard **F1 Score** (F_1) is computed as a harmonic mean of P_r and R_r.

5.2.2 Results

Table 3 shows that RIBE achieves better precision, recall, F_1, and yield for almost all relation types. The DBpedia extractor underperforms for two reasons. First, the extraction fully relies on Wikipedia anchor links for entity linking, which not only hurts the precision due to erroneous links, but also results in a low linked ratio since mentions without anchor links will not be linked. Secondly, it treats each row in an infobox table as one mention without proper chunking in the absence of anchor links. This approach hurts both precision and recall, since an extracted string value of "Sasha and Malia" for *child* not only misses the correct entities for both `Sasha` and `Malia`, but also provides false information that "Sasha and Malia" represents one single person. In contrast, RIBE identifies object mentions from infobox rows and predicts entity links even when no anchor link exists. Also, RIBE is able to consistently link to entities whose types are compatible to the target relation type. For example, we consistently link to an occupation object (*e.g.*, `Lawyer`) for *occupation* rather than to a discipline (*e.g.*, `Law`).

Relation type (Wikidata ID)	Wikidata yield	RIBE yield	+Yield (%)	Linked (%)	Precision (%)
student count (P2196)	1,325	23,440	1758.5	N/A	94.78
doctoral student (P185)	2,998	9,131	263.0	78.8	97.96
has part (P527)*	105,825	185,804	166.2	18.9	100.00
recurring date (P837)	785	1,282	126.5	89.4	95.65
instrument (P1303)	56,877	69,800	93.3	98.6	100.00
member of sports team (P54)	1,190,242	1,600,285	51.3	95.0	96.69
unmarried partner (P451)	5,263	3,108	44.5	53.1	98.97
doctoral advisor (P184)	14,300	10,688	35.2	73.9	97.12
destination point (P1444)	4,942	1,824	19.1	98.8	100.00
award received (P166)	498,505	118,509	16.3	77.4	93.18
official website (P856)	525,496	100,623	11.0	N/A	100.00
population (P1082)	695,577	53,071	4.9	N/A	100.00
sibling (P3373)	188,328	12,129	4.1	75.4	100.00
country of citizenship (P27)	2,687,600	239,798	2.4	98.6	100.00

Table 4: Comparison with Wikidata for a sample of relation types. See Table 8 in Appendix C for the full list. The column *Wikidata yield* shows the total number of relation triples in Wikidata per relation type. RIBE *yield* shows the total number of relation triples extracted by RIBE. +*Yield* represents the number of relation triples we extract that are not in Wikidata divided by Wikidata counts. Data is added to Wikidata organically by editors and the source is not limited to infoboxes. Therefore yield comparison shows that extractions from infoboxes may complement Wikidata to construct a better knowledge graph.*Linked (%)* shows that the percentage of relation triples with their objects linked to Wikidata entities ("N/A" if the object type is a literal value). $*$ We use P527 to represent *band has member* where the subject entity is a band and the object entity is a current or past member of the band.

5.3 Complement to Wikidata

We compare RIBE to Wikidata using the same subject type filter on Wikidata relation types. Table 4 shows the evaluation for a sample of relation types (see Table 8 in Appendix C for the complete list). To evaluate the quality of extra yield, we compute per-relation-type precision by randomly sampling 100 relation triples that do not exist in Wikidata (around 5.1k labels in total). The predictions are graded by human annotators and precision is computed as described in Sec. 5.2.1. RIBE achieves a significant increase in yield over Wikidata (17 out of 51 relation types have 100%+ increase), while maintaining higher than 95% precision for almost all relation types. This indicates that the extracted triples are high-quality and a critical complement to Wikidata.[16] We observe that relation types with person object type have a lower linked ratio since not all objects have corresponding entities in Wikidata (*e.g.*, children of celebrities).

5.4 Completeness of list-value relation types

A list-value relation type allows multiple objects for the same subject (*e.g.*, a person can have multiple children). In order to measure the completeness of extractions for list-value relation types, we define a completeness score C for each relation type r using a set equality by comparing the extracted set of object values for each subject entity to the ground truth set. To compute C_r, we average the completeness scores over the sampled subject entities. The "C (%)" column of Table 3 shows that RIBE consistently produces substantially more complete extractions than DBpedia does.

6 Conclusion and Future Directions

We proposed a novel system, RIBE, that extracts knowledge triples from Wikipedia infobox tables. The proposed system produces high-quality data and improves the average F_1 score over 51 relation types by 41.3% compared to a strong IBE system, DBpedia. We also empirically show the added value of the extracted knowledge with respect to Wikidata. Additionally, RIBE takes into account unlinkable entities, dramatically improving the completeness of list-value relation types.

In future work, we would like to investigate its effectiveness and robustness in a cross-lingual setting. We would like to work on Entity Discovery (Hoffart et al., 2014; Wick et al., 2013) to discover and disambiguate the unlinkable entities. We would also like to jointly model the relation extractor and the entity linker to improve the model performance.

Acknowledgments

We would like to thank Mike Cafarella, Pablo Mendes, Stephen Pulman, Chris DuBois, Lawrence Cayton, Matthias Paulik, Madian Khabsa, and Jo Daiber for providing valuable feedback and discussion, Mark Biswas for proofreading our manuscript, Thomas Semere for leading the annotation project, and Eric Chahin, Vivek Raghuram, and Eric Choi for the engineering support. We also appreciate the comments from the anonymous reviewers.

[16] Wikidata is mostly curated by human edits and therefore the Wikidata yield and RIBE yield is not directly comparable.

References

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary G. Ives. 2007. Dbpedia: A nucleus for a web of open data. In *Proceedings of the 6th International Semantic Web Conference and 2nd Asian Semantic Web Conference (ISWC/ASWC-2007)*, pages 722–735.

Sören Auer and Jens Lehmann. 2007. What have innsbruck and leipzig in common? extracting semantics from wiki content. In *ESWC*.

Razvan Bunescu and Raymond Mooney. 2007. Learning to extract relations from the web using minimal supervision. In *Annual meeting-association for Computational Linguistics*, volume 45, page 576.

M. Craven and J. Kumlien. 1999. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology, August 6-10, 1999, Heidelberg, Germany*, pages 77–86. AAAI.

S. Cucerzan. 2007. Large-scale named entity disambiguation based on wikipedia data. In *Proceedings of EMNLP-CoNLL*, volume 2007, pages 708–716.

Rajarshi Das, Arvind Neelakantan, David Belanger, and Andrew McCallum. 2017. Chains of reasoning over entities, relations, and text using recurrent neural networks. In *EACL*.

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 *KDD*.

J.R. Finkel, T. Grenager, and C. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2001. *The elements of statistical learning*, volume 1. Springer series in statistics New York, NY, USA:.

Kelvin Guu, John Miller, and Percy Liang. 2015. Traversing knowledge graphs in vector space. In *EMNLP*.

Sebastian Hellmann. 2018. Wikidata Adoption in Wikipedia. `https://lists.wikimedia.org/pipermail/wikidata/2018-December/012681.html`.

Johannes Hoffart, Yasemin Altun, and Gerhard Weikum. 2014. Discovering emerging entities with ambiguous names. In *Proceedings of the 23rd international conference on World wide web*, pages 385–396. International World Wide Web Conferences Steering Committee.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 541–550.

Mitchell Koch, John Gilmer, Stephen Soderland, and Daniel S Weld. 2014. Type-aware distantly supervised relation extraction with linked arguments. In *EMNLP*.

Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. 2009. Collective annotation of Wikipedia entities in web text. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 457–466. ACM.

Ni Lao, Tom M. Mitchell, and William W. Cohen. 2011. Random walk inference and learning in a large scale knowledge base. In *EMNLP*.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195.

Xiao Ling, Sameer Singh, and Daniel S. Weld. 2015. Design challenges for entity linking. *TACL*, 3:315–328.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1003–1011, Stroudsburg, PA, USA. Association for Computational Linguistics.

Alexander Ratner, Christopher De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. 2016. Data programming: Creating large training sets, quickly. *Advances in neural information processing systems*, 29:3567–3575.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the conll-2000 shared task chunking. In *CoNLL/LLL*.

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*, pages 455–465. Association for Computational Linguistics.

TAC. 2017. Cold start knowledge base population at tac 2017 task description. Technical report.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Michael Wick, Sameer Singh, Harshal Pandya, and Andrew McCallum. 2013. A joint model for discovering and linking entities. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 67–72. ACM.

Fei Wu and Daniel S. Weld. 2007. Autonomously semantifying wikipedia. In *Proceedings of the International Conference on Information and Knowledge Management (CIKM-2007)*, pages 41–50.

A Mention Generation Heuristics and Example Features

We present the heuristics used in mention generation of the relation extractor in Table 5 and example features in Table 6 and 7 for the relation extractor and the entity linker respectively.

B Implementation Details

In this appendix, we describe a few implementation details in addressing the noise from distant supervision and making collective entity link prediction.

B.1 Denoise Distant Supervision

Distant supervision for relation extraction assumes that every occurrence of two entities that participate in a known relation expresses the relation. However, the assumption does not always hold. For example, in the infobox table of John Lennon (https://en.wikipedia.org/wiki/John_Lennon), the object mention "Yoko Ono" appears under both Spouse(s) and Associated acts. Both occurrences of ("John Lennon", "Yoko Ono") will be labeled as positive instances of the *spouse* relation type in Wikidata, despite the second occurrence being unrelated. Therefore, we introduce a whitelist A_r of normalized infobox attributes [17] if the relation type r is vulnerable to noise. A relation mention $r(m_1, m_2)$ (e.g., the second occurrence of John Lennon and Yoko Ono) will be removed if the attribute a (*e.g.*, associated_act) is not in A_r (*e.g.*, $\{$spouse, wife, husband$\}$).

B.2 Collective Entity Linking

Locations are often presented in multiple levels, such as "San Jose, CA, USA". There are many candidate cities named "San Jose" in Wikidata. One candidate San Jose, CA has a relation *located in* [18] with California, which is a candidate for the mention "CA". Another candidate San Jose, Costa Rica does not have such a relation with the neighboring mention and therefore is less likely to be the correct entity. Similarly, if two cities are mentioned together, the candidate entities that are in the same country are more likely to be the correct prediction at the same time. This

[17] We normalize a raw infobox attribute by singularizing and converting all tokens to lower case.

[18] It is equivalent to the Wikidata relation type P131, located in the administrative territorial entity.

Heuristic	*Input example*	*Output example*
Keep anchor linked text intact	Earnest & Young	[Earnest & Young]
Split and reconstruct dates	Jan 12-14th, 2013	[2013-01-12, 2013-01-14]
Split on special characters	Monday/Tuesday	[Monday, Tuesday]
Split on stop words	Alice, and Bob	[Alice, Bob]

Table 5: Heuristics used for mention generation in the relation extractor.

Feature Type	*Feature*	*Example*
Lexical	Normalized infobox attribute of the object mention	PROP=member
Lexical	Position of the mention in the list of object values	POS=0
Lexical	Head and tail tokens of the object mention	t_0=Paul
Lexical	Window of k tokens to the left of the object mention	t-1=Lennon
Lexical	Window of k tokens to the right of the object mention	t+1=George
Lexical	Word shape of the object mention	NUM:ALL
Lexical	Type of the object mention	TYPE:LOC
Lexical	Whether all tokens in the mention are upper-cased	OBJ_UPPER
Conjunctive	Conjunction of two features	PROP=member&POS=0

Table 6: Example features used in the relation extractor.

Feature Type	*Feature*	*Example*
CandGen	Source of the candidate entity	source=anchor_link
CandGen	Match key between entity and mention phrase	match_key=alias
CandGen	Conjunctive Source and Match key	P740:s=link^mk=phrase
CandGen	Candidate Generator	P740:link_match_w_phrase
Type	Object entity type	P740:LOC
Phrase	Coarse-grained similarity score	fuzzy_score=3
Context	Entity and neighboring phrases	Q5355602^phrase1=england
Context	Overlap in page links and Wikidata triples	has_connection_P413

Table 7: Example features used in the entity linker.

also applies to relation types such as *educated at*, which expects schools as object entities. A location mention following the school name may help disambiguate.

We perform this collective entity linking approach (Cucerzan, 2007; Kulkarni et al., 2009; Ling et al., 2015) in the following way: for candidate entities e_i and e_j of two neighboring mentions, we represent the relation between the two by r_{ij}. From this we calculate the normalized score (NS) for a location phrase with token length ℓ defined as

$$\text{NS} = \begin{cases} \alpha \sum_i^\ell p_i & (\ell = 1) \\ \frac{1}{\ell} \sum_i^\ell p_i + \frac{1}{\binom{\ell}{2}} \sum_{i,j}^\ell r_{i,j} & (\ell > 1) \end{cases}, \quad (1)$$

where

$$r_{ij} = \begin{cases} 1.0 & \texttt{located\_in}(e_i, e_j) \\ 0.5 & \texttt{same\_country}(e_i, e_j) \end{cases}, \quad (2)$$

p_i is the local probability of e_i being the correct entity and α is a hyperparameter empirically set to 1.5. For $\ell > 1$, the maximum NS allowed is 2. A score larger than 1 implies a presence of either *located in* or *same country* relation. If NS is above a threshold, say 1.3, we sort by (ℓ, NS) in descending order and choose the top set of entities. While we prefer entities that are consistent with a longer phrase, the threshold accounts for the incompleteness of the relation *located in*. If there is no candidate with a NS above the threshold, we choose the candidate with the highest NS.

Conceptually the same method (with different link types r_{ij}) can be applied to all object types. However in practice we found that for non-location entities the impact was small and not worth the increased computation.

C Supplementary Experimental Results

In this appendix, we present the complete version of Table 4 in Table 8.

Relation name (Wikidata ID)	**Wikidata yield**	**RIBE Yield**	**+Yield (%)**	**Linked (%)**	**Precision (%)**
student count (P2196)	1,325	23,440	1758.5	N/A	94.78
work period (start) (P2031)	11,154	157,273	1384.6	N/A	100.00
statistical leader (P3279)	1,699	20,637	1125.1	90.6	98.08
allegiance (P945)	4,725	37,497	721.7	99.7	98.82
largest city ***	242	919	279.7	99.9	100.00
doctoral student (P185)	2,998	9,131	263.0	78.8	97.96
record label (P264)	46,900	146,259	236.7	66.4	97.20
location (P276)	52,320	128,419	216.0	95.6	99.17
has part (P527) *	105,825	185,804	166.2	18.9	100.00
winner (P1346)	28,182	52,016	156.7	97.9	100.00
spouse (P26)	91,458	155,795	137.2	25.8	98.98
end time (P582)	19,762	33,682	136.0	N/A	98.95
recurring date (P837)	785	1,282	126.5	89.4	95.65
genre (P136)	134,712	228,276	123.6	98.2	96.77
start time (P580)	20,611	31,557	121.9	N/A	100.00
residence (P551)	61,040	94,003	121.3	97.6	100.00
instrument (P1303)	56,877	69,800	93.3	98.6	100.00
location of formation (P740)	23,405	30,878	84.5	99.8	98.10
consecrator (P1598)	4,934	5,315	83.8	92.1	98.35
child (P40)	147,599	115,621	61.2	28.7	96.91
member of sports team (P54)	1,190,242	1,600,285	51.3	95.0	96.69
place of burial (P119)	88,068	53,584	49.6	80.9	100.00
conflict (P607)	92,886	73,931	49.2	98.5	98.15
dissolved (P576)	32,589	16,409	48.8	N/A	99.04
unmarried partner (P451)	5,263	3,108	44.5	53.1	98.97
place of death (P20)	637,832	441,445	41.7	96.2	97.12
musical conductor (P3300)	340	236	40.3	60.3	100.00
place of birth (P19)	1,685,695	1,218,098	37.1	96.5	100.00
doctoral advisor (P184)	14,300	10,688	35.2	73.9	97.12
parent **	141,409	79,372	33.9	48.7	100.00
family (P53)	22,245	14,590	33.6	84.6	88.78
student (P802)	12,665	3,813	26.3	68.1	100.00
destination point (P1444)	4,942	1,824	19.1	98.8	100.00
start point (P1427)	5,115	1,912	18.6	99.1	98.95
employer (P108)	247,406	61,741	17.1	90.9	95.54
member of political party (P102)	238,962	110,263	16.6	97.7	95.96
award received (P166)	498,505	118,509	16.3	77.4	93.18
religion (P140)	54,654	9,519	14.4	99.2	98.99
educated at (P69)	752,542	277,301	14.3	95.1	93.16
point in time (P585)	164,951	34,960	13.4	N/A	98.95
official website (P856)	525,496	100,623	11.0	N/A	100.00
occupation (P106)	3,624,331	539,557	8.5	93.4	98.99
inception (P571)	349,012	49,202	8.0	N/A	98.98
population (P1082)	695,577	53,071	4.9	N/A	100.00
date of birth (P569)	2,685,493	876,610	4.6	N/A	100.00
sibling (P3373)	188,328	12,129	4.1	75.4	100.00
date of death (P570)	1,309,427	315,381	3.9	N/A	99.00
country of citizenship (P27)	2,687,600	239,798	2.4	98.6	100.00
country (P17)	1,690,459	34,707	1.6	99.9	95.90
languages (P1412)	611,498	6,891	0.4	99.4	98.90
sport (P641)	689,697	5,466	0.2	100.0	100.00

Table 8: Comparison with Wikidata. The column *Wikidata yield* shows the total number of relation triples in Wikidata per relation type. RIBE *yield* shows the total number of relation triples extracted by RIBE. +*Yield* shows the number of relation triples we extract that are not in Wikidata. *Linked (%)* shows that the percentage of relation triples with their objects linked to Wikidata entities. $*$ We use P527 to represent *band has member* where the subject entity is a band and the object entity is a current or past member of the band. Note that *parent* and *largest city* in Table 8 do not currently exist in Wikidata. ** We combine relations for Wikidata predicates *father* (P22) and *mother* (P25) to form the *parent* relation type. *** We collect all the cities located in every location entity, e, in Wikidata that contains cities and use the city with the largest and latest population number as the largest city of e. This way, we construct a total of 242 *largest city* relations from Wikidata.