Extracting Quantifications of Knowledge Base Facts from Text

Anonymous EMNLP submission

Abstract

Information extraction is classically focused on extracting relations between objects, such as ⟨*Denmark, hasRegion, Hovedstaden*⟩. For many topics, texts also contain information on the count of relations, e.g., *"Denmark is divided into five administrative regions"*, which for some relations (e.g., *child*), appear as often as the actual facts. In this paper we develop a CRF-based method for extracting relation counts from text. We employ distant supervision using fact counts in the knowledge base as training data, encountering incompleteness as a new challenge wrt. classical fact extraction. We analyze linguistic particularities of cardinality information, and show that our method can achieve between 38% and 84% precision on four human-evaluated relations. We also analyze the presence of cardinality information for more than 200 relations in Wikidata.

1 Introduction

General-purpose knowledge bases such as Wikidata, DBpedia or YAGO (Vrandečić and Krötzsch, 2014; Auer et al., 2007; Suchanek et al., 2007) find increasing use in applications such as question answering, structured search or document enrichment, and their automated construction from text has received considerable attention. So far, construction techniques are focused on the extraction of fully qualified facts, but more often than not texts only contain relation cardinality information, i.e., the number of objects that stand in a relation with a certain subject, such as *"John has two children"* or *"Mary wrote 5 books"*, without mentioning the actual objects.

Extracting such relation cardinality information can hugely extend the scope of knowledge bases, thus allowing more accurate answers for queries that involve counts or existential quantification. For the child relation, for instance, simple manual patterns could reveal the existence of 178% more children from Wikipedia, than are currently contained in Wikidata (Mirza et al., 2016).

Another important use of relation cardinalities is KB curation (Paulheim, 2014; Zhang et al., 2017). KBs are notoriously incomplete, contain erroneous triples, and are limited in keeping up with the pace of real-world changes. For instance, even for a person of importance like U.S. president James A. Garfield, while the Wikipedia text mentions 7 children, Wikidata contains only 4. Similarly, DBpedia contains an erroneous child of Judy Moran called "Moran_family", leading to a total children count of 3, while all other sources speak only of 2 children. Extracting the cardinalities of relations could help addressing both issues.

Extracting relation cardinalities is more difficult than classical fact extraction for several reasons. For instance, one can observe that cardinality information can be compositional, as in the following sentences:

> *"Trump has three children with Ivana, a daughter with Marla, and a 10-year-old son with his current wife, Melania."*

Here, the total children count of 5, is split across three different predicates: *children*, *sons* and *daughter*.

Another challenge lies in the training data. Relation extraction usually relies on distant supervision, i.e., uses facts already contained in a KB as positive examples for identifying further patterns. In the case of relation cardinalities, however, knowledge bases frequently contain counts that are lower than what is correct.

Relation cardinalities are not extracted by state-of-the-art information extraction systems. ClausIE (Del Corro and Gemulla, 2013), for example extracts from the sentence *"Donald Trump has five children"* the triple ⟨*DonaldTrump, has, fiveChildren*⟩, i.e., it fails to recognize that *'five'* should be treated as parameter, not as part of the predicate. While IE methods that hinge on pre-specified relations for KB population (e.g., NELL (Mitchell et al., 2015)) can already capture numeric values for a few attributes such as ⟨*Berlin2016attack, hasNumOfVictims, 32*⟩, they are currently not able to learn them.

In this paper, we build upon the idea by Mirza et al. (2017) to use a distantly-supervised CRF classifier for identifying numbers in texts that express relation cardinalities. Our technical contributions are the following:

1. We discuss challenges that distinguish cardinality extraction from classical fact extraction.
2. We analyze various methods to obtain higher-quality training data by introducing an incompleteness-resilient distant supervision.
3. We investigate compositionality and linguistic variations in expressing relation cardinalities.
4. We analyze cardinality extraction in the large by evaluating 267 pairs of a class and a relation in Wikidata, finding that cardinality information is frequent for at least 12 of them.

2 Related Work

Advances on the automated construction of large-scale KBs have been largely influenced by prevalent relation extraction works, focusing either on structured data (Suchanek et al., 2007; Auer et al., 2007) or on unstructured contents over the web. For the latter, directions include extracting arbitrary facts without predefined schema, called Open IE (Mausam et al., 2012; Del Corro and Gemulla, 2013; Mitchell et al., 2015), and extracting triples based on well-defined knowledge base relations (Surdeanu et al., 2012; Koch et al., 2014; Palomares et al., 2016), in which the distant supervision approach is widely used (Craven and Kumlien, 1999; Mintz et al., 2009). There has also been work on reducing noise in distantly-supervised training data via learning only from positive examples (Min et al., 2013) or by expanding the knowledge base with information retrieval techniques (Xu et al., 2013).

Most relation extraction works has focused on non-numeric information. Madaan et al. (2016) explored relation extraction where one of the arguments is a number or a quantity (e.g., ⟨*Aluminium, atomicNumber, 13*⟩*)*. In general, most works on making sense of numbers in texts or semi-structured data (e.g., web tables) have been largely focused on temporal information (Ling and Weld, 2010; Strötgen and Gertz, 2010) and physical quantities or measures (Chaganty and Liang, 2016; Ibrahim et al., 2016; Neumaier et al., 2016).

In contrast, numbers that express relation cardinalities have received little attention so far. State-of-the-art Open-IE systems either hardly extract cardinality information or fail to extract cardinalities at all. While NELL, for instance, knows 13 relations about the number of casualties and injuries in disasters, they all contain only seed facts and no learned facts. The only prior works we are aware of are by Mirza et al. (2016, 2017), who use manually created patterns to mine children cardinalities from Wikipedia. They show that with 30 manually crafted patterns and simple filters it is possible to extract 86,227 children-cardinality-assertions with a precision of 94.3%, and introduce the idea of using a distant-supervision-trained CRF-based classifier for identifying numbers expressing relation quantities. In the present work, we build upon this idea, testing various hypotheses as how cardinality information can be expressed, and how shortcomings of incomplete training data can be overcome.

3 Relation Cardinalities

Inspired by Mirza et al. (2017), we define a mention of *relation cardinality* as follows: *"A cardinal number or a number-related term that characterizes the cardinality of a set of objects that stand in a specific relation with a certain subject."* For example, in *"Mary has **one** son and identical **twin** daughters,"* *'one'* and *'twin'* are the expressions we try to identify to determine the *hasChild* cardinality for Mary, which is 3.

An analysis on random numbers from Wikipedia articles revealed that around 19% numbers express relation cardinalities, most frequently for topics such as *sport* (e.g., matches played, goals scored), *creative work* (e.g., books written, seasons in an episode), *organization* (e.g., number of members) and *family relations* (Mirza et al., 2017). At present, tools such as the Stanford Named Entity (NE) tagger (Manning et al.,

Source	subjects	objects
Wikipedia articles		
cardinality information	.120	.350
names	.070	.175
Wikidata triples	.025	.030

Table 1: Fraction of persons (n=200) whose Wikipedia articles contain children cardinality information, children names, or who have children on Wikidata, and number of children per each method.

2014) only label such numbers unspecifically as NUMBER. Identifying which relations these expressions quantify would give them semantics.

Given the substantial occurrences of relation cardinalities, one may also wonder whether cardinality extraction can improve the existential coverage of KBs, i.e., the number of facts known to exist. To answer this question, we analyzed Wikipedia articles of 200 random persons, comparing the amount of existential information for the *hasChild* relation that can be retrieved by the following three methods: *(i) cardinality extraction*, where we focus on the relation cardinalities in the article; *(ii) counting names*, where we focus on the names of the children in the article; *(iii)* and *Wikidata triples*, where we count the children facts from the respective Wikidata pages. Note that the second method above corresponds to what standard relation extraction aims to achieve. As shown in Table 1, cardinality information allows to find children counts for 12% of all people, while names are only mentioned for 7%, and Wikidata contains children for only 2.5%. Similarly, with respect to the number of children in total, cardinality information allows learning of the existence of twice as many children as information extraction, and eleven times as many children as Wikidata knows of.

We conjecture that cardinality information can benefit both standard relation extraction, i.e., reducing false positives by extracting facts with high confidence only until a certain number of facts is reached, and question answering, as many questions such as *"Which US presidents were married thrice?"* only require knowledge of counts.

4 Relation Cardinality Extraction

Problem Statement Given a relation/predicate p, a subject s and a corresponding text about s, we aim to extract the *relation cardinality*, i.e., the count of $\langle s, p, * \rangle$ triples, from relation cardinality mentions in the text.

Methodology We approach the problem via sequence labeling, i.e., given a sentence containing at least one number, we employ a classifier to determine for each number in the sentence whether it is a mention of the cardinality of the relation of interest. We use CRF++ (Kudo, 2005) to build a Conditional Random Field (CRF) based classification model for each relation, taking as features the context lemmas (window size of 5) around the observed token t, along with bigrams and trigrams containing t. Note that we use `_num_` as the lemma of each cardinal number found in the text, and multi-word numbers such as *'twenty one'* are collapsed into single tokens.

The relation cardinality of a given $\langle s, p \rangle$ pair is predicted by selecting the number in the text positively annotated by the classifier, which has marginal probability–resulting from forward-backward inference–higher than 0.1. If there are several such numbers in the text, the one having the highest probability is chosen.

Distant Supervision We rely on distant supervision to generate training data. Given a knowledge base predicate p, for each entity s that appears as subject of p, we retrieve the triple count $\langle s, p, * \rangle$ from the knowledge base and a text about s. In particular, we use Wikidata as knowledge base and the Wikipedia page of each entity as text source, both in their version as of March 20, 2017.

We generate training data by annotating *candidate numbers*[1] in the text as correct cardinalities whenever *(i)* they correspond to the exact triple count and *(ii)* if they modify a noun,[2] i.e., there is an incoming dependency relation of label *nummod* according to the Stanford Dependency Parser (Manning et al., 2014). Otherwise, they are labelled as O (for Others), like the rest of non-number tokens.

Dataset We chose four Wikidata predicates that span various domains: *child* (P40), *spouse* (P26), *has part* (P527) and *contains administrative territorial entity* (P150)–for brevity henceforth called *contains admin*. While the subjects of *contains*

[1] Numbers that are not labelled as DATE, TIME, DURATION, SET, MONEY and PERCENT by the Stanford NE-tagger.

[2] This is to exclude numbers as in "*one* of the reasons..." from positive training examples.

p	$\#s$
has part	
- series of creative works	614
- musical ensemble	8,750
contains admin	6,118
child	38,496
spouse	43,668

Table 2: Number of Wikidata instances as subjects ($\#s$) of each predicate (p) in the training set.

admin, *child* and *spouse* relations are of fairly uniform type (mostly *administrative territorial entity* and *human*), the *has part* relation is used in highly diverse domains, ranging from chemical substances and groups of buildings to organizations. We decided to focus on two classes of subjects for *has part*, which are *series of creative works* (e.g., film series, novel) and *musical ensemble* (e.g., band, orchestra).

Considering only subjects of the abovementioned predicates that have links to English Wikipedia pages, we set aside 200 random subjects for each predicate as *test set*; 100 instances of each class for *has part* relation. The remaining subjects that have at least one $\langle s, p, * \rangle$ triple are used as *training set*. Furthermore, we set aside 200 random subjects per predicate from the training set as *validation set*. Table 2 reports the number of subjects ($\#s$) for each considered predicate (p) in the training set.

Evaluation We report in the first rows of Table 3, the performance of our CRF-based method (vanilla) in predicting relation cardinalities, evaluated on the validation set. While we initially wanted to use knowledge base counts for the evaluation, it turned out that these were too often too low, thus we manually annotated the validation set with the true relation counts. Moreover, whenever the predicted number and the relation count matches, we manually check whether the textual evidence, i.e., sentence containing the predicted number, truly expresses the relation of interest.

We initially built one classifier for each predicate. However, we noticed that if we use distinct classifiers for each class in *has part*, i.e. one for *creative works* and another for *musical ensemble*, the performance improved considerably, particularly for *creative works* (.222 vs .372 F1-score). The method works reasonably well for *creative works* and *contains admin*, with .372 and .325 F1-scores, respectively. For *musical ensemble* and *spouse*, on the other hand, both precision and recall suffer, resulting in an overall performance of only around 2% F1-score.

We next discuss major limitations of the vanilla approach as revealed by the qualitative evaluation, and how to tackle them.

5 Improving Relation Cardinality Extraction

5.1 Training Data Quality

Unlike training data for normal fact extraction, which is generally highly correct (e.g., YAGO claims 95% precision (Suchanek et al., 2007)), taking triple counts found in knowledge bases as ground truth generally gives wrong results. For example, our manual annotation of the validation set for *child* shows that about 50% of the KB counts are incorrect wrt. the knowledge one can derive from Wikipedia texts.

Mirza et al. (2017) showed that manually generated training data can hugely boost performance, however, obtaining sufficient quantities of manually annotated data is generally costly. We see several avenues to tackle the training data quality issue.

Incompleteness-resilient Distant Supervision Triple counts in the knowledge base are often lower than what is correct, but rarely too high. During the training data generation, these incorrect counts will generate spurious negative examples. For example, recalling President Garfield, for whom Wikidata knows only 4 out of his 7 children, the number "seven" in the sentence *"In 1858, he married Lucretia; they would have* ***seven*** *children..."* on his Wikipedia page[3] would be labelled as negative example, leading to a lower probability for numbers appearing in similar contexts to be labelled as correct cardinalities.

Since there is no way to know whether higher numbers in the text are actually positive examples, one possible approach is to treat them as neither positive nor negative examples, but simply remove them from the training set. We test two variations of this approach:

- Ignore $n > c$, i.e., we remove sentences that only contain numbers (n) that are higher than the triple count (c).

[3] https://en.wikipedia.org/wiki/James_A._Garfield

	has part						contains admin			child			spouse		
	creative works			musical ensemble											
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
combined	*.238*	*.208*	*.222*	*.030*	*.023*	*.026*									
vanilla	**.421**	**.333**	**.372**	**.016**	**.011**	**.013**	**.660**	**.216**	**.325**	**.200**	**.159**	**.177**	**.028**	**.017**	**.021**
Training Data Quality															
ignore $n > c$	-.04	0	-.02	+.02	**+.02**	**+.02**	-.09	+.01	-.01	+.01	+.03	+.02	-.00	+.01	+.00
$c < n \leq c+1$	-.01	0	-.00	-.00	0	0	0	0	0	+.01	+.01	+.01	-.01	-.01	-.01
$c < n \leq c+2$	+.00	**+.01**	**+.01**	+.01	+.01	+.01	0	0	0	+.02	+.02	+.02	+.00	**+.01**	**+.01**
$c < n \leq c+3$	-.00	+.01	+.01	+.01	+.01	+.01	0	0	0	+.03	**+.03**	**+.03**	-.01	0	-.00
exclude freq. n	**+.07**	-.02	**+.01**	**+.03**	+.01	**+.02**	**+.04**	-.01	-.00	**+.10**	+.06	**+.08**	-.03	-.02	-.02
$n \leq 1$	+.01	+.01	+.01	+.44	+.05	+.09	+.03	0	+.00	**+.07**	+.04	**+.05**	+.03	-.01	-.00
$n \leq 2$	**+.06**	+.02	**+.04**	**+.70**	+.05	**+.09**	+.14	-.01	+.01	+.16	-.07	-.03	**+.97**	0	**+.01**
$n \leq 3$	+.02	-.09	-.06	+.58	+.02	+.05	**+.16**	-.01	**+.01**	+.60	-.14	-.13	-.03	-.02	-.02
top 25%	0	0	0	0	0	0	0	0	0	-.01	-.01	-.01	-.00	0	-.00
50%	**+.01**	0	**+.00**	-.00	0	-.00	0	0	0	-.01	-.01	-.01	-.01	-.01	-.01
75%	0	0	0	-.00	0	-.00	0	0	0	-.00	-.01	-.01	-.00	0	-.00
best train	**.525**	**.323**	**.400**	**.714**	**.056**	**.104**	**.800**	**.209**	**.332**	**.377**	**.278**	**.320**	**1.00**	**.046**	**.087**
Compositionality															
comp	-.06	+.01	-.01	0	0	0	**+.06**	**+.18**	**+.20**	**+.01**	**+.01**	**+.01**	-.33	0	-.00
Linguistic Variance															
transform	**+.06**	**+.13**	**+.11**	**+.09**	**+.03**	**+.05**	0	0	0	-.01	-.01	-.01	-.15	**+.02**	**+.03**
transform 'a'	-.12	-.02	-.06	-.26	-.01	-.01	-.11	-.04	-.06	-.12	-.06	-.08	-.67	-.01	-.02
best final	**.587**	**.449**	**.509**	**.800**	**.087**	**.157**	**.855**	**.386**	**.532**	**.384**	**.290**	**.330**	**.846**	**.063**	**.116**

Table 3: Evaluation results on the validation set.

- Ignore $c < n \leq c+d$, i.e., we remove sentences that only contain numbers slightly higher than the triple count, for values of d between 1 and 3.

Excluding Uninformative Numbers The more frequent a certain number occurs in a text, the more probable it is to occur in various contexts. As a way to give the classifier less noisy training examples, one might wish to filter out frequently occurring numbers irrespective of whether they match the triple count or not. Specifically, we experiment with labeling numbers that occur more than 5 times in a text as negative examples.

By Benford's law, lower numbers are more frequent than higher numbers. As a very simple heuristic, we thus also experiment with excluding all n, $1 \leq n \leq 3$ from the training examples.

Filtering Ground Truth Instead of taking the triple counts for all subjects of a predicate as ground truth, one might trade size for quality. We rank the subjects according to their *popularity*, i.e., the number of triples/facts about them stored in the knowledge graph. We then experiment with using only the 25%, 50% and 75% most popular subjects as training data.

5.2 Compositionality

We observed that cardinalities for *contains admin* were often mentioned as a composition of several numbers, e.g., *"The Qidong county has **4** subdistricts, **17** towns and **3** townships under its juridiction."* This phenomenon is also observed for *child*, as exemplified at the beginning of Section 3.

In this paper, we focus on number compositionality when a sequence of numbers occurs in the same sentence. In training data generation, if the sum of such a number sequence is equal to the triple count, we label all numbers in the sequence as positive examples.

In the prediction step, we predict the relation cardinality by summing up consecutive numbers labelled as positive with sufficient probabilities by the classifier. To avoid predicting the wrong cardinality in *"He had **four** children: **two** sons and **two** daughters"* we check the number sequence as follows: for a predicted number p labelled as positive, if the sum of the following numbers, that are also labelled as positives, is equal to p, we simply choose p as the correct relation cardinality. In the previous example, our method will predict *four* as the children count instead of *eight*.

5.3 Linguistic Variance

Our initial motivation was to make sense of the so far ignored large fraction of numbers that express relation cardinalities. However, we noticed quickly that relation cardinalities are frequently also expressed with other concepts related to numbers such as *trilogy* or *duo*.

We used used the *relatedTo* relation in ConceptNet (Speer and Havasi, 2012) for collecting terms related to numbers. We split the terms into two groups, those having Latin/Greek prefixes[4] and those not having them. For the first group, we generated a list of Latin/Greek prefixes, e.g., *tri-*, *quart-*, and a list of possible suffixes, e.g., *-logy*, *-et*. We manually checked the latter group to select only terms that were strongly associated with cardinalities, e.g., *twin*, *thrice* and *dozen*.

In a pre-processing step, a Latin/Greek number found in the text is represented with only its suffix as the lemma, and labelled as a positive example if its prefix corresponds to the relation count. For example, when we found *'triplet'* in the text, its lemma will be converted to `_plet_` and it will be labelled as a positive example if the relation count is equal to 3. For other terms, we simply replace them with the correct terms containing cardinal numbers, e.g., *twin* → *two children*, *thrice* → *three times* and *dozen* → *twelve*.

We also observed that the relation cardinality of *one* is frequently represented with indefinite articles, for instance, *"They had **a** son together"* or *"It has **a** residential community and 7 villages under its adminstration."* Therefore, we also experiment with converting indefinite articles *a* and *an* in the test/validation set into *one*.

6 Analysis

6.1 Evaluation on the Validation Set

We performed an ablation study to identify the impact of each idea from above wrt. the vanilla approach. The results are reported in Table 3, based on the same evaluation methodology used in Section 4.

[4] http://phrontistery.info/numbers.html

Training Data Quality Ignoring numbers larger than KB counts was found to slightly improve the performance, except for *contains admin*. We presume the reason for this is that Wikidata is already highly complete for this relation. For other relations, the varying degree of deviation d that improves the performances hints at how many $\langle s, p, * \rangle$ triples per subject s are usually missing from the knowledge graph, i.e., $d = 3$ for *child*, and $d = 2$ for *creative works* and *spouse*. For *musical ensemble*, ignoring all higher numbers is the best approach, which suggests that Wikidata is remarkably incomplete for that relation.

Excluding numbers frequently occurring in the text turns out to considerably improve precision (except for *spouse*), for instance by 10% for *child*. Excluding low numbers has a similar effect, although the effect appears very much dependent on the nature of the predicates, i.e., the average number of $\langle s, p, * \rangle$ triples that are often mentioned as cardinality assertions for the observed predicate p in the text about s. For instance, when excluding $n \leq 1$ is the best setting for *child*, then that means that two children are frequently mentioned in texts, hence, excluding $n \leq 2$ would filter more positive than negative examples.

Somewhat surprisingly, taking smaller but more complete subsets for training did not have any effect on performance. We conjecture that for these instances, a more complete knowledge base is offset by longer and thus more noisy articles.

In Table 3, we report the extraction performance after our attempts to improve the training data quality (best train) by using the corresponding best setting (shown in bold) for each predicate. The *best train* scores are then used to further show the impact of tackling compositionality and linguistic variance discussed below.

Compositionality and Linguistic Variance The results on tackling the compositionality and linguistic variance issues shed further light on the nature of each relation. Cardinality assertions for *contains admin* are very often compositional, as shown by the improvement of 20% in F1-score, seldom for *child* with 1% F1-score increase, and not at all for the others.

Instead, the other relations benefited from considering concepts related to numbers as candidates for relation cardinality. We observe significant improvements of both precision and recall for *has part*, and of recall for *spouse*. This approach allows the extraction method to infer the relation count from terms such as *'pentalogy'*, *'duo'* and *'(married) twice'*.

Transforming all indefinite articles *'a'* and *'an'* into *'one'* in the test data, in turn, results in a great increase of false positives, and reduces precision considerably.

The final performance of our extraction method for each relation on the validation set is shown in the last row (best final) of Table 3. The method works quite well for *contains admin*, *spouse* and *musical ensemble* with 85.5%, 84.6% and 80% precision scores respectively. The low recall for *musical ensemble* and *spouse* reflects the rarity of cardinality assertions containing cardinal numbers (or number-related terms) for those relations. Average performance with 50.9% F1-score on *has part* for *creative works* might be due to the comparably small training data set. Meanwhile, we attribute an observed lower precision on *child* to three factors:

1. The classifier often confuses the number of children with, for instance, number of siblings, spouses, or (political) terms served.
2. The number-of-children assertions found in the text (about a person) are actually about someone else, e.g., his/her parent or sibling.
3. The total number of children can be inferred from numbers mentioned in several sentences, as in *"John married Jane in 1983. They have **two** children together. After their divorce in 1995, he married Jamie, with whom he has **two** sons and **one** daughter."*

6.2 Evaluation on the Test Set

We also evaluated the performance of our method on the test data, which contains crowd-annotated 200 random entities per relation. We used the CrowdFlower[5] platform for annotating *(i)* whether the number of objects could be inferred from the Wikipedia page of a certain subject, and *(ii)* what that number was, taking in each case the majority vote among three crowdworkers. Quality was ensured via unambiguous test questions. It turns out that the task was not trivial, as on the random entities, annotators voted unanimously in only 83% of cases. Frequent reasons for disagreement were for instance for *has part*, when different granularities like *"3 seasons and 12 episodes"* were mentioned, or when for a band, a vocalist, two guitarists and a drummer were mentioned, but it was left unclear whether these were all members.

In Table 4, we report the performance of our method on the crowd-annotated dataset. The recall (RCE, R) was computed by using the total number of subjects of which the crowd could infer their object cardinality from Wikipedia articles. Our method could extract cardinality information with precision (RCE, P) ranging from 40% to 62.5%.

We also report in the next columns the percentage of subjects (%subject) for which *(i)* our method could extract the relation counts correctly (RCE), *(ii)* Wikidata contains at least on fact in the respective relation, and *(iii)* the crowd workers said one could infer the relation count by any means from the Wikipedia article. As one can see, for *contains admin* and *child* the percentage of subjects of which our method succeed in extracting the cardinalities is reasonably close to the ones of Wikipedia. For *creative works*, *musical ensemble* and *spouse*, the large gap stems from the facts that Wikipedia articles more often mention the individual objects, which allows crowd workers to infer the cardinality by counting, a technique that is currently not accessible by our method.

In the existential knowledge increase column we report the impact of relation cardinality extraction towards enlarging the existential knowledge of KBs, in this case Wikidata. For *creative works* and *child*, the number of facts known to exist increased significantly, by 17.3 and 7.6 times respectively. Meanwhile, for *musical ensemble*, Wikidata usually already contains the ensemble member names, so extracting cardinality information does not help much.

7 Large-scale Run of RCE

We collected all Wikidata properties that were not asserted to be single-value[6], had a functionality degree ($\#triples/\#subjects$) of less than 0.98 (Galárraga et al., 2015), and were used by at least 500 subjects, obtaining 267 properties in total.

[5] https://www.crowdflower.com/

[6] Properties having the property constraint type https://www.wikidata.org/wiki/Q19474404

p	RCE P	RCE R	RCE F1	%subject RCE	%subject Wikidata	%subject Wikipedia	existential knowledge increase (Wikidata+RCE) / Wikidata
has part							
- creative works	.545	.279	.369	.120	.020	.550	17.3
- musical ensemble	.400	.026	.049	.020	.280	.770	1.1
contains admin	.571	.308	.400	.020	.060	.065	1.8
child	.625	.750	.682	.070	.020	.095	7.6
spouse	.500	.026	.050	.005	.020	.019	1.8

Table 4: Evaluation results on the test set; RCE denotes *Relation Cardinality Extraction*.

For each property/relation, we set aside the 200 of the 400 most popular entities as test set, while using the rest (limited to 10k most popular entities) as training data. Note that we only considered entities of the most frequent type for each class, e.g., *human* for *sibling*, to ensure domain homogeneity. We then ran our Relation Cardinality Extraction (RCE) system for each property, using the setting we assume to generally work well for all relations (vanilla + ignore $c < n \leq c + 2$ + exclude freq. n + exclude $n \leq 1$). We evaluated the precision wrt. the triple counts for the entities in the test set, assuming that for the most popular entities, these are usually correct.

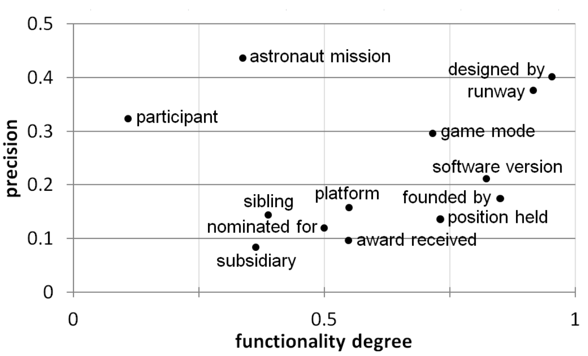

Figure 1: Precision results on some notable Wikidata relations, along with their corresponding functionality degrees.

There were a total of 147 for which RCE could identify relation cardinalities with more than 5% precision. While some are spurious results due to low variance, in Figure 1 we show some properties where the results were manually found to be not mere coincidences. These properties are used, for instance, for humans (e.g., *sibling*, *award received*), games/software (e.g., *designed by*, *software version*), companies (e.g., *founded by*, *subsidiary*) and transportation-related buildings (e.g., *platform*, *runway*). Our method also achieves an impressively high precision of 97.8% on *contains settlement*, which is a relation similar to *contains admin*.

8 Conclusion

We have introduced the problem of extracting relation cardinalities from text, and discussed the challenges that set it apart from standard information extraction. There are several avenues to extend this work. On the technical side, the present work does not consider instances with no facts in training (due to their overwhelming proportion), and is thus not suited to predict zero cardinality (like Angela Merkel having no children).

Furthermore, compositionality is only explored within sentences, while in reality it appears also spread over multiple sentences. Taking this even further, one might even look at multiple sources, which may have different timestamps, and use techniques from truth discovery and data fusion to retrieve most likely values in the case of conflicts.

A third direction is to go towards constraints and statistical reasoning. Ordinal number like in *"His second wife"* are ignored by our method, but are valuable clues as they set lower bounds on relation cardinalities. Similarly, the number of brothers and sisters should add up to the number of siblings, having 80 band members is uncommon, or sports teams normally have fewer coaches than players. Learning such constraints, or exploiting them in the consolidation part of relation cardinality extraction, could be fruitful to further improve precision and recall of the present method.

References

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. *DBpedia: A nucleus for a web of open data*. Springer.

Arun Chaganty and Percy Liang. 2016. How much is 131 million dollars? putting numbers in perspective with compositional descriptions. In *ACL*. pages 578–587.

Mark Craven and Johan 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*. pages 77–86.

Luciano Del Corro and Rainer Gemulla. 2013. ClausIE: clause-based open information extraction. In *WWW*. ACM, pages 355–366.

Luis Galárraga, Christina Teflioudi, Katja Hose, and Fabian M Suchanek. 2015. Fast rule mining in ontological knowledge bases with amie+. *VLDB Journal* 24(6):707–730.

Yusra Ibrahim, Mirek Riedewald, and Gerhard Weikum. 2016. Making sense of entities and quantities in web tables. In *CIKM*. pages 1703–1712.

Mitchell Koch, John Gilmer, Stephen Soderland, and Daniel S. Weld. 2014. Type-aware distantly supervised relation extraction with linked arguments. In *EMNLP*. pages 1891–1901.

Taku Kudo. 2005. CRF++: Yet another CRF toolkit. *Software available at http://crfpp. sourceforge.net* .

Xiao Ling and Daniel S Weld. 2010. Temporal information extraction. In *AAAI*. volume 10, pages 1385–1390.

Aman Madaan, Ashish Mittal, G Ramakrishnan Mausam, Ganesh Ramakrishnan, and Sunita Sarawagi. 2016. Numerical relation extraction with minimal supervision. In *AAAI*. pages 2764–2771.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. *ACL (System Demonstrations)* pages 55–60.

Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. 2012. Open language learning for information extraction. In *EMNLP*. pages 523–534.

Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. 2013. Distant supervision for relation extraction with an incomplete knowledge base. In *NAACL*. pages 777–782.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *ACL*. pages 1003–1011.

Paramita Mirza, Simon Razniewski, Fariz Darari, and Gerhard Weikum. 2017. Cardinal virtues: Extracting relation cardinalities from text. In *ACL 2017 Short Papers (to appear, available on arXiv)*.

Paramita Mirza, Simon Razniewski, and Werner Nutt. 2016. Expanding Wikidatas parenthood information by 178%, or how to mine relation cardinalities. *ISWC Posters & Demos* .

Tom M. Mitchell, William W. Cohen, Estevam R. Hruschka Jr., Partha Pratim Talukdar, Justin Betteridge, Andrew Carlson, Bhavana Dalvi Mishra, Matthew Gardner, Bryan Kisiel, Jayant Krishnamurthy, Ni Lao, Kathryn Mazaitis, Thahir Mohamed, Ndapandula Nakashole, Emmanouil Antonios Platanios, Alan Ritter, Mehdi Samadi, Burr Settles, Richard C. Wang, Derry Tanti Wijaya, Abhinav Gupta, Xinlei Chen, Abulhair Saparov, Malcolm Greaves, and Joel Welling. 2015. Never-ending learning. In *AAAI*. pages 2302–2310.

Sebastian Neumaier, Jürgen Umbrich, Josiane Xavier Parreira, and Axel Polleres. 2016. Multi-level semantic labelling of numerical values. In *ISWC*. pages 428–445.

Thomas Palomares, Youssef Ahres, Juhana Kangaspunta, and Christopher Ré. 2016. Wikipedia knowledge graph with DeepDive. In *ICWSM*. pages 65–71.

Heiko Paulheim. 2014. Identifying wrong links between datasets by multi-dimensional outlier detection. In *WoDOOM*. pages 27–38.

Robert Speer and Catherine Havasi. 2012. Representing general relational knowledge in conceptnet 5. In *LREC*.

Jannik Strötgen and Michael Gertz. 2010. Heideltime: High quality rule-based extraction and normalization of temporal expressions. In *SemEval Workshop*. pages 321–324.

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. YAGO: a core of semantic knowledge. *WWW* pages 697–706.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance multi-label learning for relation extraction. In *ACL*. pages 455–465.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM* 57(10):78–85.

Wei Xu, Raphael Hoffmann, Le Zhao, and Ralph Grishman. 2013. Filling knowledge base gaps for distant supervision of relation extraction. In *ACL (short paper)*. pages 665–670.

Jiawei Zhang, Jianhui Chen, Junxing Zhu, Yi Chang, and Philip S Yu. 2017. Link prediction with cardinality constraint. In *WSDM*.