Numerical Relation Extraction with Minimal Supervision

Aman Madaan[*]
Visa Inc.
amadaan@visa.com

Ashish Mittal[*]
IBM Research
mittalashish61@gmail.com

Mausam
IIT Delhi
mausam@cse.iitd.ac.in

Ganesh Ramakrishnan
IIT Bombay
ganesh@cse.iitb.ac.in

Sunita Sarawagi
IIT Bombay
sunita@cse.iitb.ac.in

Abstract

We study a novel task of *numerical relation extraction* with the goal of extracting relations where one of the arguments is a number or a quantity (*e.g.*, atomic_number(Aluminium, 13), inflation_rate(India, 10.9%)). This task presents peculiar challenges not found in standard Information Extraction (IE), such as the difficulty of matching numbers in distant supervision and the importance of units. We design two extraction systems that require minimal human supervision per relation: (1) NumberRule, a rule based extractor, and (2) NumberTron, a probabilistic graphical model. We find that both systems dramatically outperform MultiR, a state-of-the-art non-numerical IE model, obtaining up to 25 points F-score improvement.

Introduction

While there is a long history of relation extraction systems in the NLP literature (*e.g.*, (ARPA 1991; Soderland 1999; Hoffmann et al. 2011; Riedel et al. 2013)), almost all information extractors have concentrated on relations in which the arguments are non-numerical. These include real world entities or objects, or other attributes that are usually expressed in words, such as color and job title. Several extractors do deal with specific numerical regular expression types such as dates, while some extract the age of individuals, but almost none have focused on *numerical relations*, *i.e.*, relations involving general numeric arguments such as population, area, atomic number, inflation rate, or boiling point. Numerical relations form a significant subset of relations in many fields, including science, current affairs, geography, and healthcare; extraction of numerical information from text is an important Information Extraction (IE) problem requiring research attention.

This is especially true since numerical relations present several peculiarities and challenges not found or less prevelant in standard IE. Firstly, and probably most importantly, modern IE systems are based on distant supervision, in which the presence of entities from a database relation in a sentence is indicative of the presence of that relation in that sentence. The signal from distant supervision becomes much weaker for numerical relations since there can be a much larger number of reasons why a certain number is present in the sentence. This renders distant supervision based non-numerical extractors less effective for numerical relations. In our early experiments, MultiR (Hoffmann et al. 2011), a state-of-the-art IE system, obtained an F-score of under 20, hardly acceptable for real tasks. Secondly, numbers have units and their semantics is important. Thirdly, numbers may be written at different rounding levels necessitating partial matching techniques. Lastly, numerical relations allow for sentences which describe the *change* in the argument value from the last measurement, instead of the argument value itself.

In response, we develop two numerical relation extractors that incorporate these observations . Both extractors expect minimal human supervision in the form of the unit of the relation and up to four keywords indicative of that relation. Our first system, NumberRule, is a rule-based extractor that looks for occurrences of specific numerical relation based patterns that explicitly mention the given keywords. Our second system, NumberTron, goes beyond the given keywords to learn new keywords and patterns and can also leverage any existing background Knowledge base (KB).

We evaluate our extractors on the task of extracting numerical indicators (e.g., inflation rate) for countries. We compile a knowledge-base using geopolitical data from World Bank and learn extractors for ten numerical relations. We find that NumberTron obtains a much higher recall at a slightly higher precision as compared to NumberRule. Both systems massively outperform MultiR model (and its simple extensions) obtaining 17–25 point F-score improvements.

We release our code[1] and other resources for further research. Overall, we make the following contributions in this paper:

- We define and analyze the task of numerical relation extraction. Our analysis highlights stark differences in this task compared to standard IE.
- We design NumberRule, a rule-based system that looks

[*]Most work was done when the authors were graduate students at IIT Bombay.

[1]Available at `http://www.github.com/NEO-IE`

for pre-defined patterns with specific keywords to extract a numerical relation.

- We design NumberTron, an extension of MultiR for numerical relation extraction that can learn new patterns while also exploiting other features specific to our task.
- We compile a knowledge-base and a test set of 430 sentences for this task from the geopolitical domain. Our experiments reveal that NumberTron obtains much higher recall and F-score than NumberRule, and both systems outperform the MultiR model as well as a recall oriented baseline by wide margins.

Related Work

Relation extraction from text has a long history going back to the Message Understanding Conferences (MUC) (ARPA 1991; 1998). Early systems were rule-based and supervised approaches were developed later (*e.g.*, (Freitag 1998; Zhao and Grishman 2005; Bunescu and Mooney 2005)). Supervised techniques require huge amounts of labeled data per relation making them less scalable to many relations.

To reduce human input, several *distant supervision* approaches have been developed where training dataset is automatically labeled by aligning an unsupervised corpus with a knowledge-base of facts (Craven and Kumlien 1999). Early approaches hypothesized a distant supervision assumption: if a sentence has the two entities from a fact in the KB then that sentence is a positive datapoint for the KB relation (Mintz et al. 2009).

The original idea has since been refined to explicitly handle the noise due to the distant supervision assumption. Riedel et al (2010) relaxed the assumption that *every* such sentence is a positive training data by using multi-instance learning. Subsequently, MultiR and MIML-RE (Hoffmann et al. 2011; Surdeanu et al. 2012) allowed the model to learn multiple relations between the same pair of entities. Recent extensions obtain better negative examples (Min et al. 2013), allow for the KB and corpus to be incomplete (Ritter et al. 2013), and improve extraction via better entity detection, coreference and linking (Koch et al. 2014).

NumberTron is a high precision adaptation of MultiR that incorporates signals from units, pre-specified keywords, number features, and more to reduce noise of matching numbers to a KB. The recent extensions to MultiR are orthogonal to our task, and are equally applicable to NumberTron.

Numerical Relations: Most relation extraction literature has focused on non-numerical relations, with a handful of exceptions like age, year of birth, etc. (TACKBP 2014). Davidov and Rappaport (2010) use bootstrapping and pattern learning for extracting properties like height and width. The key system that scales to generic numerical relations is LUCHS (Hoffmann, Zhang, and Weld 2010). It used distant supervision style matching from Wikipedia infoboxes (and Web lists) to create over 5,000 relation extractors, which included numerical relations.

For numerical relations LUCHS used Gaussian features that facilitate partial matching between numbers. Since it mostly matched arguments to text in the *same* article, this form of partial matching was sufficient for its task. But this won't be effective for us, since an entity and a quantity co-occurring in general text is an extremely weak signal for a relation. Nguyen & Moschitti (2011) and Intxaurrondo et. al (2015) also extract some numerical attributes using ideas similar to LUCHS.

Quantities in NLP: Early work on formal semantics addressed quantities in language (Montague 1973; Hurford 1975). Most recent work on numbers has concentrated on specific subdomains like temporal expressions (e.g., (Pustejovsky et al. 2003), (Do, Lu, and Roth 2012)). Some application areas such as Web search (Banerjee, Chakrabarti, and Ramakrishnan 2009) and solving science and arithmetic questions (Kushman et al. 2014; Hosseini et al. 2014) have also observed the importance of numbers. Quantities have also been recognized as an important part of textual entailment systems (e.g., (MacCartney and Manning 2008; Roy, Vieira, and Roth 2015)).

Quantities are typed via units, for example, 'm/s' in "330 m/s", and '$' in "$2,000". Extracting units from text can be challenging and recently Sarawagi & Chakrabarti (2014) developed a context free grammar based number unit extractor. It extracts numbers, their multipliers, and units and normalizes them into a number and its SI unit. We use this extractor in our systems.[2]

Numerical Relation Extraction

Our goal is to extract a set of binary relations R such that second argument (arg2) of the relation is a quantity with a given unit and the first argument (arg1) is an entity from a given semantic class. For example, from the sentence "*Aluminium is a chemical element in the boron group with symbol Al and atomic number 13*", we wish to extract the relation atomic_number(Aluminium, 13). Our focus is geopolitical relations such as inflation_rate(India, 11%) and land_area(USA, 2,959,054 square miles).

In alignment with existing research on IE we do not expect annotated training data per relation. We take two kinds of inputs for learning classifiers: (1) we allow an NLP expert to provide a few keywords that are indicative of each relation, and (2) we can also make use of a background KB that has facts about these relations. In addition, we assume access to a large unsupervised text corpus. We first describe challenges that numerical relations bring to the task of IE.

Weak Signal from Distant Supervision: Distant supervision techniques build on the insight that if two arguments appear together in a sentence, there is a good chance they may express the relation. However, since quantities can appear in far more contexts than typical entities, distantly supervised training data becomes much more noisy to be useful. For example, we can imagine a handful of relations between "Bill Gates" and "Microsoft" (founder, CEO, etc), but it is much harder to list possible relations between, say, India and 11%. This situation is far worse for small whole numbers that appear unit-less or with popular units (e.g., percent) than for quantities like 11.42143 or 330 m/sec.

[2]Available at `https://github.com/ssprojects/UnitTagger`

Sentence	Test
The estimated population of Australia is about 36.25 million people.	-
The estimated population density of Australia is 36.25 million people **per sq km**.	1
The estimated population of Australia **increased** *by about 36.25 million people.*	2
The estimated population of **urban** *Australia is about 36.25 million people.*	3
The estimated **adolescent** *population of Australia is about 36.25 million people.*	3
The estimated populations in 2014 are Australia, **100 million** *and New Zealand, 36.25 million.*	4

Table 1: NumberRule outputs total_population(Australia, 36.25 million) only in the first sentence. The second column is test number that fails for other sentences. The input keyword is "population".

This problem can also be seen in regular IE. E.g., "John Smith" may map to many entities leading to noisy distant supervision matches. However, in regular IE every KB column has relatively few such entities, unlike in numerical IE.

Match Mines: A related manifestation of the same problem is match mines – when a certain KB entry causes an unprecedented number of matches in the corpus. This typically happens when the arg1 is a popular entity and arg2 is a small whole number (e.g., China and 3). In our dataset, a few sentences were responsible for 21% matches. Often these were score tables of games (e.g., soccer) between teams representing two countries. We ought to discard such sentences even if they have candidate mentions.

Partial Matching: Unlike standard entity-entity relations, wherein the second entity rarely or never changes, numbers can change rapidly (e.g., inflation of a country). Moreover, the same quantity can be expressed using different number of significant digits in different sentences. These necessitate partial matching techniques within distant supervision.

Unit Processing: Units act as types for numbers. The same quantity may be expressed with different units (e.g., 20 kms and 12.4 miles). A numerical extractor needs to perform unit conversions for correct matching and extraction.

Change Words: Often sentences, especially news stories, express the change in a value instead of, or in addition to, the actual value itself. E.g., "*Amazon stock price increased by $35 to close at $510.*", can easily confuse an extractor whether the stock price is $35 or $510. It is important to detect *change words* (e.g., 'increase') for accurate extraction.

Relation/Argument Scoping: Additional modifiers to arguments or relation words may subtly change the meaning and confuse the extractors. E.g., extracting from "*rural* literacy rate of India", or "literacy rate of *rural* India" will not be accurate when extracting India's literacy rate. Such structures are common in numerical IE, since numerical relations can be easily re-scoped for different parts of an entity.

Importance of Keywords: In contrast to all the aforementioned challenges, there is one observation that makes a large subset of numerical relations easier. Many numerical relations are mediated by one or a handful of keywords (usually nouns). For example, sentences expressing "inflation rate", "GDP", "life expectancy" would often use these keywords; patterns not using these keywords would be uncommon. While this is not true for all numerical relations, it is often true – we exploit this observation in designing and learning keyword features for effective extraction.

NumberRule

We now present NumberRule, a generic rule-based numerical relation extractor, which uses insights from the previous section to develop rules to obtain high precision. The only relation-specific supervision to NumberRule is a small list of keywords per relation. For example, the total population of a country relation may have a keyword 'population'.

The basic NumberRule system first creates a dependency parse of a given sentence. It uses collapsed typed dependencies as obtained from the Stanford parser (Manning et al. 2014). It then performs Named Entity Recognition (NER) to identify candidate arg1s in the sentence based on matching with the expected type of arg1 for the relation. It then finds the shortest path in the dependency parse between a candidate arg1 and a number. Finally, it checks for the occurrence of one of the pre-specified relation keywords either on the shortest path, or on an immediate connection to any token on the shortest path through an amod, nn, vmod or advmod edge. If it finds the keyword it extracts the relation between candidate arg1 and the number.

Of course, this basic NumberRule system will have very low precision since it does not incorporate numerical-relation specific insights from the previous section. We improve the precision of this system by adding four tests. An extraction is outputted only if all four tests succeed. First, we test whether the unit after the number is equivalent to the input unit for the relation arg2. The unit extractor directly gives us this information (Sarawagi and Chakrabarti 2014). Second, we look for change words on the shortest path and if one is found we discard the extraction. This allows us to remove sentences that express change in numeric value instead of the absolute value. The change words used in NumberRule are 'change', 'up', 'down', 'grow', 'increase', 'decrease', 'surge', and 'rise'. Third, we discard any extraction where the arg1 or the keyword has a modifier via an amod, nn, vmod or advmod edge. This gets rid of errors due to a misplaced argument or relation scoping.

If an extraction passes these three tests, we make one final check. In case there are multiple arg1s and (or) multiple valid number-unit pairs in the sentence, we output only one extraction per arg1 – the one that is closest to it in the dependency parse. If multiple valid numbers are closest, we pick the leftmost one to the right of the entity. Table 1 presents several examples that illustrate situations where these tests are able to avoid common errors.

The NumberRule system is not a learning system and does not go beyond the given keywords for extracting a relation.

In the next section, we present NumberTron, which can learn new phrases and also identify bad given keywords for a relation using distant supervision.

NumberTron

NumberTron uses a graphical model like MultiR (Hoffmann et al. 2011) for relation extraction, but with several differences in detail to address the unique challenges posed by numerical extraction.

The Graphical Model

Unlike MultiR which creates a graph for each entity-pair, NumberTron creates a graph for each entity. This allows it to reason about multiple numeric values associated with an entity jointly. At the highest level, the graphical model maintains z nodes indicating that a sentence expresses a certain relation, and n nodes denoting that a numeric quantity must be extracted (with an entity) for a given relation aggregated over multiple sentences. Join potentials between n and z express this aggregation. Node potentials at z express sentence-level features, which can learn new patterns relevant for a given relation. We now describe the model in more detail.

For each entity e, let Q_e denote the distinct numbers with unit[3] that are observed in sentences S_e that mention entity e. For each $q \in Q_e$, let $S_{e,q} \subseteq S_e$ denote the sentences that mention e and q. For each entity e and relation r, our graphical model contains one binary random variable n_q^r for each $q \in Q_e$ and one binary random variable z_s^r for each $s \in S_{e,q}$. For any $S_{e,q}$ we only consider those candidate relations r where q is tagged with a unit compatible with r's.

Each z_s^r variable is associated with a node potential ψ_s^r computed from a set of features ϕ_s and associated relation specific parameters θ^r as $\psi_s^r(z_s^r = 1) = \exp(\theta^r \phi_s)$. The $\mathbf{n}$ and $\mathbf{z}$ nodes are constrained by ψ^{join} potentials to ensure that $\mathbf{n}$ variables are only under sufficient support from $\mathbf{z}$ variables and to include agreement among close-by numbers (more on this later). There are no parameters attached to these potentials. Thus, the joint distribution over labels of sentences that contain the entity e is

$$\Pr(\mathbf{z}, \mathbf{n} | S_e, Q_e, \theta) = \frac{1}{\mathbb{Z}} \prod_{r \in \mathcal{R}} \prod_{s \in S_{e,q}} \psi_s^r(z_s^r) \psi^{join}(\mathbf{n}^r, \mathbf{z}^r)$$

where Z is the normalization constant. For the node potential ψ_s^r we use all the features in (Mintz et al. 2009) derived from words and POS tags on the dependency paths connecting the entity and the number. In addition, we create a special category of features called 'Keyword Features' corresponding to the pre-specified relation keywords (also used in NumberRule). We also create special 'number features' as follows: first we convert each number unit pair to its canonical SI unit. We then add features characterizing the scale and type of the number like: is the number whole or fractional, is the number between 0 and 100, is the number in thousands, millions, billions, etc. The Mintz features are general enough to capture change words and thus we do not express them explicitly.

[3]Our unit tagger converts all unit variants like 'mile', 'km' to a canonical SI unit (in this case, 'meter').

Parameter Learning

We learn parameters θ using distant supervision and a perceptron-like training algorithm (Collins 2002). We start with an unlabeled text corpus and a KB of seed numerical triples. We first describe how we use the KB to get supervision. We cannot use exact match of numbers in the corpus to the KB. Instead we perform a soft match as follows. For each entity e, number q with unit u in the corpus, let $\text{KB}_{e,u}$ denote the triples (e, r, v) in the KB with relation r's unit u. We set an n_q^r to "1" if q is within $\pm\ \delta_r\%$ (set to 20%, obtained via cross validation) of v for some triple (e, r, v) in $\text{KB}_{e,u}$ and one of the pre-specified keywords of r appears in any of the sentences containing q. Else we set an n_q^r to false if $\text{KB}_{e,r}$ is non-empty. A z_s^r variable takes the label of its linked n_q^r variable. All unset n, z variables are removed from the graphical model. Let $\bar{\mathbf{n}}_e, \bar{\mathbf{z}}_e$ denote the assigned variables. Later, we experimentally compare with other methods of using the KB for supervision.

We use the Collins perceptron algorithm to train the θ parameters using the $\bar{\mathbf{n}}, \bar{\mathbf{z}}$ assignments over several entities as labeled data. The training loop needs inference to find $\hat{\mathbf{n}}, \hat{\mathbf{z}} \equiv \text{argmax}_{\mathbf{n}, \mathbf{z}} \Pr(\mathbf{n}, \mathbf{z} | S_e, Q_e; \theta)$. We design an efficient inference algorithm that can run in one pass over large training datasets. For each sentence s, we first set $\hat{z}_s^r = 1$ for any r whose $\psi_s^r(1)$ is largest (i.e., $= \max_{r' \in \mathcal{R}} \psi_s^{r'}(1)$) and greater than zero. We then assign the $\mathbf{n}$ variables based on the $\hat{z}_s$ variables and the constraints imposed by the ψ^{join} potentials. We experimented with the following definitions of the join potentials:

- **Simple OR**: $\hat{n}_q^r$ is set to one if and only if there exists any $s \in S_{e,q}$ such that $\hat{z}_s = r$.
- **Atleast-K**: $\hat{n}_q^r$ is set to one iff at least k fraction of $s \in S_{e,q}$ have $\hat{z}_s = r$. We use $k = 0.5$ for our experiments.
- **Agreeing-K**: We wish to additionally enforce that two proximal number nodes should either both be zero or both be one. In this scheme we start with the Atleast-K assignment $\hat{\mathbf{n}}$ and choose a central value c (similar to the true value of the relation). We set to zero any n_q^r outside a band of $\pm\delta\%$ of c, and others are set to 1. We choose the central value c for which $\hat{n}_c^r = 1$ and which causes the smallest number of $\hat{n}_q^r$s that were 1 and are flipped to zero.

Extraction: We perform sentence level extraction. For each sentence, we identify each entity, quantity pair $s = (e, q)$ and calculate the score $\psi_s^r(1)$ for each candidate relation r that matches the unit of q in the sentence. We predict label r if the min-max normalized log score is greater than some threshold α. We use cross validation to set α = 0.90.

Discussion

NumberTron differs from MultiR in a number of ways. NumberTron's graph is made per-entity instead of per entity-pair. Moreover, it fixes the assignment of $\mathbf{z}$ variables based on pre-specified keywords, whereas MultiR only labeled $\mathbf{n}$

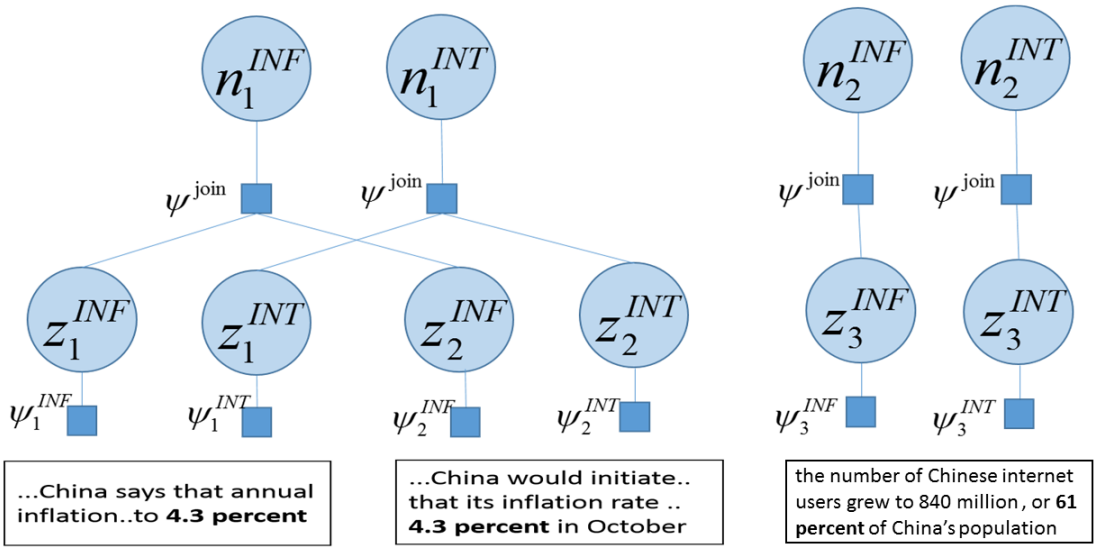

Figure 1: NumberTron Graphical Model for Entity *China*. Three sentences mention two percentages, 4.3% and 61%, represented as n_1 and n_2 respectively. INF denotes inflation rate, and INT is used for percent internet penetration. Each sentence has z nodes (for each relation) denoting that the sentence is expressing the relation. Each n node denotes that the quantity n is an accurate argument for the relation. Multiple z nodes offer support for n nodes via join potentials. z nodes are linked to number nodes if the quantities are within $\delta_r\%$ of each other.

nodes based on KB match; it kept **z** floating and assigned them in an EM step based on ψ^{join} potentials. Since chance matches with numbers is very high, MultiR-style labeling results in a lot of noise. NumberTron's modification would likely result in higher quality matching. This also makes NumberTron's inference algorithm simpler. Finally, NumberTron's join potentials are more general than simple-OR and require a much strong support for each fact than in MultiR. This mitigates the problem of weak signal in distant supervision described earlier.

NumberTron also incorporates additional features. Number features are specific to the task of numerical relation extraction. Keyword features, on the other hand, are in response to the observation of importance of keywords in this task. NumberTron uses unit-normalization to handle unit variations while matching. It also allows partial matching of numbers for scenarios where quantities are mentioned at different rounding levels.

NumberTron heuristically cleans the training set by removing sentences with change-words. This allows it to create a cleaner distantly supervised data. Textual pattern features can naturally deal with presence of change words by assigning low weight to those as long as training data is clean. NumberTron also removes two KB entries that have similar values and units but different relations for the same entity. Finally, it removes extremely long sentences from the text corpus, since they are usually responsible for the match mines.

Experiments

We evaluate NumberTron and NumberRule, and compare it with two baselines: a high recall most frequent class baseline and a version of MultiR (Hoffmann et al. 2011) that we signficantly improved for numerical relations. We also analyze the differences between NumberTron and NumberRule, and perform ablation tests to assess the usefulness of our feature set and the value of distant supervision.

Training Corpus We train on the TAC KBP 2014 corpus (TACKBP 2014) comprising roughly 3 million documents from NewsWire, discussion forums, and the Web.

Knowledge Base We compile our KB[4] from data.worldbank.org. This data has 1,281 numerical indicators for 249 countries, with over 4 million base facts. Our experiments are on ten of these relations listed in Table 2. We pick these relations since they form a diverse and challenging set. The units do not trivially determine the relation since we have two relations with 'percent' unit, and three with 'US dollar' unit. The Population relation is unitless, causing every unitless number to be a possible candidate, thus attracting significant noise. The range of values for Internet users and Inflation is overlapping and both are often small percentages, causing them to be confounded with arbitrary relations not in our set.

Test Set The test corpus is a mix of 430 sentences from the TAC corpus and sentences from Web search on relation name. Web search was needed since TAC corpus did not have many positive examples for some of the relations. Table 3 shows the number of instances per relation in this corpus and also the number of negatives – sentences that do not have any extraction from our set of relations – grouped by relations of the same unit.

[4]Available at `https://github.com/NEO-IE/numrelkb`

Relation	Keywords
Internet User %	internet
Land Area	area, land
Population	population, people, inhabitants
GDP	gross, domestic, GDP
CO_2 emission	carbon, emission, CO2, kilotons
Inflation	inflation
FDI	foreign, direct, investment, FDI
Goods Export	goods, export
Life Expectancy	life, expectancy
Electricity Production	electricity

Table 2: Pre-specified keywords

Unit tagging In addition to the standard NLP pipeline, we pre-processed both the training and test corpus using the unit tagger (Sarawagi and Chakrabarti 2014) – it extracts numbers, their multipliers, and units and normalizes them into a number and its SI unit.

Keywords Table 2 lists the 1-4 keywords we provided per relation as input. We mostly used the signficant words in the name of the relation and did not carefully tune the keywords to assess the robustness of our systems.

Relation	**Units**	**Positive**	**Negative**
Land Area	Sq. Km	57	17
Population	-	51	300
Inflation	percent	51	84
Internet Users	percent	15	
FDI	$ (USD)	10	35
GDP	$ (USD)	8	
Goods Export	$ (USD)	11	
Life Expectancy	year	15	34
Electricity Production	kWh	13	6
CO_2 Emissions	kiloton	8	16

Table 3: Test corpus statistics: The 3rd column is number of instances per relation and the 4th is the number of "none-on-the-above" ($\perp$) grouped by relation of the same unit.

Baseline Algorithms

We compare NumberRule and NumberTron with two baselines: a recall oriented prior-based baseline and a numerical adaptation of MultiR.

Recall-Prior baseline For each unit it predicts the relation with the highest *test* prior ignoring the "none-of-the-above" class. For example, as per Table 3, all numbers tagged with "USD" unit will be labeled 'Goods exported' since after ignoring the "none-of-the-above" class it is the most frequent class. Naturally this baseline will have perfect recall for relations that do not conflict with another relation on units.

Adapting MultiR for Numerical Relations For fair comparison we substantially improved MultiR[5] extractor for numerical relations. We provided it with the same unit tagger as in our algorithms for identifying and normalizing numbers and units. Similar to NumberTron, we used the units to narrow down candidate relations during training and testing. We also added our partial matching (using $\pm\delta_r\%$) technique in distant supervision. Finally, we provided it keyword-based features for fair comparison against other systems. We call this MultiR++.

[5]Downloaded from `https://github.com/jgilme1/MultirExperiments` commit 0b465a74dc49b298

System	**Precision**	**Recall**	**F1 Score**
MultiR++	50.00	31.75	38.84
Recall-Prior	28.18	86.19	42.47
NumberRule	59.30	53.60	56.30
NumberTron	60.93	66.92	63.78

Table 4: Aggregate results. NumberTron outperforms all.

Relation	NumTron F1	NumRule F1
FDI	0	50.00
Life Expectancy	68.96	69.50
Internet Users	55.73	54.54
Electricity Prod.	50.00	62.50
GDP	57.14	42.80
CO_2 Emissions	47.61	53.30
Inflation	88.40	56.25
Goods export	75.00	35.20
Population	49.99	60.30
Land Area	57.44	52.22

Table 5: Per relation F1 for NumberRule and NumberTron

Comparison of different methods

The aggregate results of the four systems on our complete test set are presented in Table 4. These results use the best settings of NumberTron, which are described in the ablation study section. We observe that NumberTron provides the best overall precision-recall values, followed closely by NumberRule. Recall-Prior baseline has very high recall but a much lower precision. As expected, we find that the main merit of a statistical method like NumberTron over a rule-based method like NumberRule is in the increased recall, which jumps from 53.6% to 67%. The simple prior-based base line has very poor precision, but the recall is high because it never predicts the "none-of-the-above" class. The performance of MultiR++ is surprisingly poor (without keyword features the F-score was under 20). This is likely because of additional enhancements in NumberTron that are missing in MultiR++, such as number features, fixed assignment of $\mathbf{z}$ variables, and general join potentials.

Analysis

We further analyze the strengths and weaknesses of NumberTron and NumberRule. NumberRule's missed recall is primarily because of not having a keyword on the dependency path. An illustrative example is: "*Turkey's central bank said Wednesday it expects the annual inflation rate to reach 6.09 percent at the end of 2009 , lower than the official target of 7.5 percent.*". From this sentence, NumberRule does not extract inflation_rate(Turkey, 6.09 percent), because the keyword 'inflation' is not on the shortest dependency path between Turkey and 6.09 (Turkey $\xrightarrow{poss}$ bank $\xrightarrow{nsubj}$

Distant Supervision	Simple OR			Atleast-K			Agreeing-K		
	P	R	F1	P	R	F1	P	R	F1
KB	43.24	50.93	46.54	40.05	53.93	45.97	35.20	44.52	39.35
Keywords	43.35	73.22	54.46	43.69	73.62	54.83	45.97	70.80	55.74
KB + Keywords	61.56	64.96	63.21	60.93	66.92	63.78	63.46	60.21	61.79

Table 6: Comparison of various configurations for NumberTron

Features	P	R	F1
Mintz features only	22.85	36.86	28.21
Keyword features only	51.24	52.55	51.89
Mintz + Keyword	47.10	39.04	42.71
Mintz + Number	17.80	35.03	23.67
Keyword + Number	45.15	69.70	54.80
Mintz + Key. + Num.	*60.93*	*66.92*	*63.78*

Table 7: Ablation tests of feature templates for NumberTron

said $\xrightarrow{ccomp}$ expects $\xrightarrow{xcomp}$ reach $\xrightarrow{dobj}$ percent $\xrightarrow{num}$ 6.09). On the other hand, since NumberTron combines evidences from multiple features, it outputs this extraction – several features like number's range, presence of 'inflation' and 'rate' in the context and three different dependency path patterns fire for NumberTron.

Table 5 lists the F-scores of the two systems for each relation. By and large NumberTron wins on recall, and has performance within 10-15 points of NumberRule. However, for FDI relation, NumberTron does not output a single extraction! This is because sentences expressing this relation are rare in our training corpus.

On Goods and Population, NumberRule has an unusually weaker recall. Both these relations are well represented in the training corpus making it easier for NumberTron to learn. Moreover, NumberRule's test 4 significantly reduces recall for these – many test sentences mention multiple values for the same entity-relation in a sentence, from which NumberRule extracts only the first. An (abridged) example is "*Annual average inflation for Lithuania fell to 7.9 percent in July from 8.7 percent in June and 9.4 percent in May.*".

Finally, population relation is unusual in that NumberRule has high recall and low precision, and NumberTron is exactly reverse. This was because one of the pre-keywords was 'people'. This is a generic word and led to many errors for NumberRule. On the other hand, NumberTron powered by the KB learns low weight for this keyword, and improves precision, but this also hurts recall.

Ablation Study for NumberTron

We now report the experiments that help us in identifying the best configurations for NumberTron. Earlier, we describe three choices for the design of ψ^{join} potential – Simple OR, Atleast-K, and Agreeing-K. Moreover, we implemented three different approaches for labeling the training data ($\bar{\mathbf{z}}_e$ variables) – (1) heuristically label all sentences with the right unit, keyword and entity as positive label, (2) distant supervision using KB, and (3) both keyword-based and KB-based distant supervision. This results in nine different configurations. Table 6 presents a comparison.

We verify from this experiment that standard distant supervision offers very weak signal for numerical extraction – results on KB only are not very good. Keywords are crucial, and KB in conjunction with keyword-based labeling adds significant value. We also learn that Atleast-K provides marginally better results than Simple OR. The Agreeing-K potential that enforces numbers to be within a band of δ is not as good, possibly because in the early stages of training, when the parameters are not well-trained, this is too severe a restriction. Overall we select Atleast-K in conjunction with KB + Keywords-based labeling as the best setting.

We also study the impact of the various features in node potentials of NumberTron. These include the original Mintz features (Mintz et al. 2009), keyword-based features, and various number-specific features as discussed in Section . Table 7 presents the results. We find that by themselves the large set of Mintz features confuses the classifier; keyword features are much more effective. Number features substantially improve F1 in the presence of keywords. Combining all three yields the best performance.

Conclusions and Future Work

We present the first detailed study of the task of numerical relation extraction, in which one of the arguments of the relation is a quantity. Our preliminary analysis reveals several peculiarities that make the task differently challenging from standard IE. We employ these insights into a rule-based system, NumberRule, that can extract any numerical relation given input keywords for that relation. We also develop NumberTron, an extension of MultiR, which employs novel task-specific features and can be trained via distant supervision or other heuristic labelings.

By aggregating evidence from multiple features, NumberTron produces much higher recall at comparable precision compared to NumberRule. Both systems vastly outperform baselines and non-numerical IE systems, with NumberTron yielding almost 25 point F-score improvement.

A key limitation of our research is lack of temporal modeling – many numerical relations change over time. In the future, we wish to extract numerical relations along with their temporal scopes. Temporal identification will likely improve the effectiveness of distant supervision too.

Acknowledgments

We thank Dan Weld for helpful discussions. We thank the anonymous reviewers for insightful comments. This work was partially supported by an IBM Faculty Award to Dr. Ganesh Ramakrishnan, Google language understanding and knowledge discovery focused research grants to Dr. Mausam, and a KISTI grant to Dr. Mausam.

References

ARPA. 1991. *Proc. 3rd Message Understanding Conf.*

ARPA. 1998. *Proc. 7th Message Understanding Conf.*

Banerjee, S.; Chakrabarti, S.; and Ramakrishnan, G. 2009. Learning to rank for quantity consensus queries. In *ACM SIGIR*.

Bunescu, R. C., and Mooney, R. J. 2005. A shortest path dependency kernel for relation extraction. In *HLT/EMNLP*.

Collins, M. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *EMNLP*.

Craven, M., and Kumlien, J. 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*, 77–86.

Davidov, D., and Rappoport, A. 2010. Extraction and approximation of numerical attributes from the web. In *ACL*.

Do, Q.; Lu, W.; and Roth, D. 2012. Joint inference for event timeline construction. In *EMNLP-CoNLL*.

Freitag, D. 1998. Toward general-purpose learning for information extraction. In *COLING-ACL*.

Hoffmann, R.; Zhang, C.; Ling, X.; Zettlemoyer, L. S.; and Weld, D. S. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *ACL: HLT*.

Hoffmann, R.; Zhang, C.; and Weld, D. S. 2010. Learning 5000 relational extractors. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, 286–295.

Hosseini, M. J.; Hajishirzi, H.; Etzioni, O.; and Kushman, N. 2014. Learning to solve arithmetic word problems with verb categorization. In *EMNLP*.

Hurford, J. R. 1975. *The Linguistic Theory of Numerals*. Cambridge Univ Press.

Intxaurrondo, A.; Agirre, E.; de Lacalle, O. L.; and Surdeanu, M. 2015. Diamonds in the rough: Event extraction from imperfect microblog data. In *NAACL HLT 2015*, 641–650.

Koch, M.; Gilmer, J.; Soderland, S.; and Weld, D. S. 2014. Type-aware distantly supervised relation extraction with linked arguments. In *EMNLP*.

Kushman, N.; Zettlemoyer, L.; Barzilay, R.; and Artzi, Y. 2014. Learning to automatically solve algebra word problems. In *ACL*.

MacCartney, B., and Manning, C. D. 2008. Modeling semantic containment and exclusion in natural language inference. In *COLING*.

Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J. R.; Bethard, S.; and McClosky, D. 2014. The stanford corenlp natural language processing toolkit. In *ACL Demonstrations*.

Min, B.; Grishman, R.; Wan, L.; Wang, C.; and Gondek, D. 2013. Distant supervision for relation extraction with an incomplete knowledge base. In *HLT-NAACL*.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *ACL*.

Montague, R. 1973. *The proper treatment of quantification in ordinary English*. Springer.

Nguyen, T. T., and Moschitti, A. 2011. End-to-end relation extraction using distant supervision from external semantic repositories. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA - Short Papers*, 277–282.

Pustejovsky, J.; Castaño, J. M.; Ingria, R.; Sauri, R.; Gaizauskas, R. J.; Setzer, A.; Katz, G.; and Radev, D. R. 2003. Timeml: Robust specification of event and temporal expressions in text. In *New Directions in Question Answering, Papers from 2003 AAAI Spring Symposium*, 28–34.

Riedel, S.; Yao, L.; McCallum, A.; and Marlin, B. M. 2013. Relation extraction with matrix factorization and universal schemas. In *HLT*, 74–84.

Riedel, S.; Yao, L.; and McCallum, A. 2010. Modeling relations and their mentions without labeled text. In *ECML (PKDD)*.

Ritter, A.; Zettlemoyer, L.; Mausam; and Etzioni, O. 2013. Modeling missing data in distant supervision for information extraction. *TACL* 1:367–378.

Roy, S.; Vieira, T.; and Roth, D. 2015. Reasoning about quantities in natural language. *TACL* 3:1–13.

Sarawagi, S., and Chakrabarti, S. 2014. Open-domain quantity queries on web tables: annotation, response, and consensus models. In *ACM SIGKDD*.

Soderland, S. 1999. Learning information extraction rules for semi-structured and free text. *Machine Learning* 34(1-3):233–272.

Surdeanu, M.; Tibshirani, J.; Nallapati, R.; and Manning, C. D. 2012. Multi-instance multi-label learning for relation extraction. In *EMNLP-CoNLL*.

TACKBP. 2014. *Text Analysis Conference Knowledge Base Population*.

Zhao, S., and Grishman, R. 2005. Extracting relations with integrated information using kernel methods. In *ACL*.