Identification and Verification of Simple Claims about Statistical Properties

Andreas Vlachos and **Sebastian Riedel**
Department of Computer Science
University College London
{a.vlachos,s.riedel}@cs.ucl.ac.uk

Abstract

In this paper we study the identification and verification of simple claims about statistical properties, e.g. claims about the *population* or the *inflation rate* of a country. We show that this problem is similar to extracting numerical information from text and following recent work, instead of annotating data for each property of interest in order to learn supervised models, we develop a distantly supervised baseline approach using a knowledge base and raw text. In experiments on 16 statistical properties about countries from Freebase we show that our approach identifies simple statistical claims about properties with 60% precision, while it is able to verify these claims without requiring any explicit supervision for either tasks. Furthermore, we evaluate our approach as a statistical property extractor and we show it achieves 0.11 mean absolute percentage error.

1 Introduction

Statistical properties are commonly used to describe entities, e.g. *population* for countries, *net_value* for companies, *points_scored* for athletes, etc. Claims about such properties are very common in news articles and social media, however they can be erroneous, either due to author error or negligence at the time of writing or because they eventually become out of date. While manual verification (also referred to as fact-checking) is conducted by journalists in news organizations and dedicated websites such as `www.emergent.info`, the volume of the claims calls for automated approaches, which is one of the main objectives of computational journalism (Cohen et al., 2011; Flew et al., 2012).

In this paper we develop a baseline approach to identify and verify simple claims about statistical properties against a database. The task is illustrated in Figure 1. Given a sentence, we first identify whether it contains a claim about a property we are interested in (`population` in the example), which entity it is about and the value claimed (Lesotho and 2,000,000 respectively). We then proceed to verify the value claimed in text for the property of this entity against the value known in a knowledge base such as Freebase and return a score reflecting the accuracy of the claim (absolute percentage error in the example).

Text: Lesotho, a landlocked enclave of South Africa, has a population of nearly 2 million and covers an area slightly smaller than the U.S. state of Maryland.

Entity: Lesotho
Property: `population`
Value claimed in text: 2,000,000

Value in knowledge base: 2,193,843
Absolute percentage error: 0.09

Figure 1: Claim identification and verification.

Claim identification is essentially an instance of information extraction. While it would be possible to develop supervised models, this would require expensive manual data annotation for each property of interest. Instead, we follow the distant supervision paradigm (Craven and Kumlien, 1999; Mintz et al., 2009) using supervision obtained by combining triples from a knowledge base and raw text. However, statistical properties are more challenging in applying the distant supervision assumption than relations between named entities due to the fact that the numerical values are often approximated in text, as in the example of Figure 1. Consequently, linking the values mentioned in text with those in the knowledge base is not trivial and thus it is not straightforward to generate training instances for the property of interest.

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 2596–2601,
Lisbon, Portugal, 17-21 September 2015. ©2015 Association for Computational Linguistics.

To address this issue, we propose a distantly supervised claim identification approach that relies on approximate instead of exact matching between values in text and the knowledge base. In experiments on 16 statistical properties about countries from Freebase we show that our approach identifies simple statistical claims with 60% precision, while it is able to verify these claims without requiring any explicit supervision for this task. In developing our approach, we also evaluate it as a statistical property extractor achieving 0.11 mean absolute percentage error. The code and the datasets developed are publicly available from `https://github.com/uclmr/simpleNumericalFactChecker`.

2 Claim identification algorithm

Our approach to claim identification relies on discovering textual patterns between an entity and a numerical value used to express the property of interest. For example, the first, second and fourth patterns in Table 1 express the *population* property, and we would like our approach to select them to identify claims about this property.

During training, we assume as input a set of entity-value pairs from the knowledge base for the property of interest (top-right part of Table 1) and a set of textual patterns (bottom-left part), each associated with entity-value pairs (bottom-right part). The patterns and the entity-value pairs associated with them are obtained by processing raw text, which we discuss in Section 3.

The key difficulty compared to other applications of distant supervision is that numerical values are often approximated in text. Thus, instead of looking for patterns that report the exact values for each entity, we develop an approach for finding the patterns that predict the values well. Intuitively, the algorithm ranks all text patterns according to how well they predict the values for the property at question and then greedily selects them till the accuracy of the aggregate predictions by the selected paterns stop improving. To compare the predicted entity-value pairs $\hat{EV}$ against the property entity-value pairs EV_{prop} we use the mean absolute percentage error (MAPE):

$$MAPE(EV_{prop}, \hat{EV}) = \frac{1}{|E|} \sum_{e \in E'} \frac{|v_e - \hat{v}_e|}{|v_e|} \quad (1)$$

Note that only the values predicted for entities in both EV_{prop} and $\hat{EV}$ (denoted by E') are taken into account in this equation, thus in calculating MAPE for the pattern "X has _ inhabitants" from Table 1 against the entity-values for *population* only the two values present are considered. MAPE is commonly used in measuring forecasting accuracy of algorithms in finance (Hyndman and Koehler, 2006). Unlike mean absolute error or mean squared error it adjusts the errors according to the magnitude of the correct value.

Initially (line 1) the algorithm decides on a default value (v_{def}) to return for the property at question among three options: the mean of the training values, their median or zero. The criterion for the choice is which one results in a better MAPE score on the training data. We refer to this prediction as the `InformedGuess`. This default value is used when predicting (lines 16-29) in case there are no values for an entity in the patterns selected, e.g. if only the pattern "X has _ inhabitants" is selected and the prediction for Iceland is requested.

Following this, the patterns are ranked according to the MAPE score of their entity values with respect to the entity values of the property at question (lines 2-4). We then iterate over the patterns in the order they were ranked. For every pattern, we add it to the set of patterns used in predicting (lines 8-9), and evaluate the resulting predictions using MAPE with respect to the training values (line 10). If MAPE is increased (predictions become worse), then we remove the newly added pattern from the set and stop. Otherwise, we continue with the next pattern in the queue.

In experiments with this algorithm we found that while it often identified useful patterns, sometimes it was misled by patterns that had very few entities in common with the property and the values of those entities happen to be similar to those of the property. For example, the pattern "_ tourists visited X" in Figure 1 has only one entity-value pair ("France:68,000,000") and the value is very close to the *population* value for "France". To ameliorate this issue, we adjusted the MAPE scores used in the ranking step (line 4) according to the number of values used in the calculation using the following formula:

$$\text{adjusted}MAPE = \frac{c}{c+N} MAPE \quad (2)$$

where N is the number of values used in calculating MAPE in Equation 1 and c is a parameter that regulates the adjustment. Lower c puts more importance on the number of values used, thus leading the algorithm to choosing patterns assessed with more values, and thus more reliably.

population	France:66,028,467, Russia:143,700,000, Iceland:325,600
the population of X is _	France:66,000,000, Russia:140,000,000, Iceland:325,000
X's population is estimated at _	France:66,030,000, Russia:145,000,000
X's inflation rate is _	France:0.9, Iceland:4.0
X has _ inhabitants	Russia:140,000,000, Iceland:300,000
_ tourists visited X	France:68,000,000

Table 1: Property and text patterns associated with entity-value pairs.

Algorithm 1: Claim identification algorithm

```
Input: Entity-values for property
       $EV_{prop} = \{(e_1, v_1), (e_2, v_2), \ldots\}$,
       patterns $P = \{p1, p2, \ldots\}$,
       entity-values for pattern $p$: $EV_p$
Output: Selected patterns $P_{sel}$
$v_{def}$ = InformedGuess ($EV_{prop}$)
priorityQueue $Q = \emptyset$
foreach pattern $p \in P$ do
    push ($Q$, ($p$, MAPE ($EV_{prop}$, $EV_p$)))
$P_{sel} = \emptyset$
$mp$ = MAPE ($EV_{prop}$, predict ($E, P_{sel}$) )
while $Q \neq \emptyset$ do
    pattern $p$ = pop ($Q$)
    $P'_{sel}$ = $P_{sel} \cup \{p\}$
    $mp'$ = MAPE ($EV_{prop}$, predict
    ($E, P'_{sel}$))
    if $mp' > mp$ then
        break
    else
        $mp = mp'$
        $P_{sel} = P'_{sel}$
function predict(entities $E$, patterns
$P_{sel}$)
    $\hat{EV} = \emptyset$
    foreach $e \in E$ do
        $sum = 0$
        $count = 0$
        foreach $p \in P_{sel}$ do
            if $(e, v) \in EV_p$ then
                $sum+ = p\{e\}$
                $count+ = 1$
        if $count > 0$ then
            $\hat{EV} = \hat{EV} \cup (e, sum/count)$
        else
            $\hat{EV} = \hat{EV} \cup (e, v_{def})$
    return $\hat{EV}$
```

3 Data collection

To evaluate the claim identification approach developed we compiled a dataset of statistical properties from Freebase. We downloaded a snapshot[1] of all instances of the *statistical_region* entity type with all their properties with their most recent values, keeping only those were from 2010 onwards. From those we selected the 16 properties listed in Table 2, each property having values for 150-175 regions (mostly countries).

To collect texts from which the text patterns between entities and numerical values will be extracted we downloaded documents from the web. In particular, for each region combined with each property we formed a query consisting of the two and submitted it to Bing via its Search API. Following this we obtained the top 50 results for each query, downloaded the HTML pages corresponding to each result and extracted their textual content with BoilerPipe (Kohlschütter et al., 2010). We then processed the texts using the Stanford CoreNLP toolkit (Manning et al., 2014) and from each sentence we extracted textual patterns between all the named entities recognized as locations and all the numerical values. Two kinds of patterns were extracted for each location and numerical value: surface patterns (as the ones shown in Table 1) and lexicalized dependency paths.

This pattern extraction process resulted in a large set of triples consisting of a region, a pattern and a value. Different sentences might result in triples containing the same region and textual pattern but different value. Such variation can arise due to either the approximations of values in text or due to the pattern being highly ambiguous, e.g. "X is _". We distinguish between the two by requiring each region-pattern combination to have appeared at least twice and its values to have stan-

[1] Data was collected in May 2014.

dard deviation less than 0.1. In this case, then the region-pattern value is set to the mean of the values it is encountered with, otherwise is removed.

4 Information extraction experiments

We first evaluate our approach as a statistical property extractor for two reasons. First, while our main goal is to develop a claim identification approach, there is no data for this task to evaluate, thus making development difficult. On the other hand, we can evaluate statistical property extraction in a straightforward way, thus facilitating development and parameter tuning. Second, the algorithm described learns such an extractor, thus it is of interest to know its performance.

We split the values collected from Freebase into 2/3 for training and 1/3 for testing, ensuring that all regions are present in both datasets. The accuracy is evaluated using MAPE. When using adjusted MAPE we set the parameter c for each property using 4-fold cross-validation.

The performance of Algorithm 1 using the unadjusted MAPE was 0.72 averaged over all properties. Using the adjusted version this was greatly improved to 0.49. We also evaluated the `InformedGuess` prediction which returns the same value for all regions (it chooses the value that performs best among the mean, the median and 0), and its overall MAPE was 0.79. Recalling that Algorithm 1 returns the `InformedGuess` in case no pattern is found for an entity, we also evaluate the performance without returning a value for such entities, thus ignoring them in the evaluation. In that case the performance with unadjusted MAPE improves to 0.17 but 10% coverage, while with adjusted MAPE it improves to 0.11 with 43% coverage. Best performances were achieved for relations such as *population* which have a wide range of values that is well separated from the rest, while percentage rates were usually harder for the opposite reason. Thus we conclude that the algorithm with adjusted MAPE selects better patterns for each property that are encountered more frequently, which is important for the main goal of this paper, claim identification.

5 Claim identification and verification

We now evaluate our approach on claim identification. For each property, we run Algorithm 1 using adjusted MAPE and the parameter c as chosen in the experiments of the previous section to select patterns expressing it. We then process all texts and if a sentence contains one of the selected patterns between an entity and a value, it is returned for manual inspection as shown in Figure 1.

Freebase property	claims	precision
consumer_price_index	116	0.93
cpi_inflation_rate	464	0.92
diesel_price_liter	212	1.00
fertility_rate	307	0.99
gdp_growth_rate	39	0.31
gdp_nominal	308	0.98
gdp_nominal_per_capita	415	0.20
gni	413	0.62
gni_per_capita	795	0.49
health_expenditure	197	0.99
internet_users_%	93	0.00
life_expectancy	581	0.45
population	1583	0.9
population_growth_rate	1377	0.11
renewable_freshwater	105	1.00
undernourishment	87	0.13
OVERALL	7092	0.60

Table 2: Claim identification results.

The claims returned were labeled by the authors of the paper as correctly or incorrectly identified according to the following guidelines. A claim is extracted correctly only if both the entity and the value are extracted correctly and the sentence expresses the property at question. E.g. a claim identified in a sentence containing a country and its correct GDP growth rate without stating it as such (the same percentage rate can be true for multiple statistical properties) is considered incorrect. Furthermore, we considered claims referring to past measurements (e.g. results of a past census) to be correctly identified.

Results for each property are shown in Table. 2. Overall precision was 60% over 7,092 statements, and it varied substantially across properties. Perfect precision was found for claims of *renewable_freshwater* for which one textual pattern was responsible for all the claims identified and it was correct. On the other hand, the zero precision for claims of *internet_user_%* was due to identifying correctly sentences listing countries and their respective values for this property but not identifying the country-value pairs correctly. More representative of properties with precise claim identification was *population*, for which the relatively few errors were due to the patterns learned not being able to distinguish between different types of population e.g. general vs working population. On the other hand, the claims for *gni_per_capita* had low accuracy because they were confused with those of *gdp_nominal_per_capita*, as their values tend to be relatively close. The claims identified and annotated manually are attached to our submission. Finally, some errors are due to the algorithm being constrained to extract a claim considering only the text pattern between the entity and the value, thus ignoring parts of the sentence that might be relevant. For example, the pattern "the population of X is _" is generally reliable, but in the sentence "The population of Tajikistan is 90 % Muslim" it extracts a claim incorrectly.

The verification stage of the simple claims we extract is rather simple; we just score the claims according to the absolute percentage error of the value claimed in text with respect to the value in known in Freebase. In the process of labeling the claims identified we did a qualitative analysis of the claims with high error. We found cases where our algorithm flagged cases of out of date estimates of populations used, e.g. the webpage `http://www.economywatch.com/world_economy/bolivia`[2] states that the population of Bolivia is 9 million, while it is estimated to be above 10 million.

6 Discussion - Related work

As explained, we tackle claim identification as an instance of information extraction, and propose a baseline able to perform both tasks. However, it is important to distinguish between them. In claim identification we are interested in all claims about a property, even inaccurate ones; in information extraction on the other hand, and especially its formulation as knowledge base population, we are interested in the accurate claims only, since extracting inaccurate ones will lead to erroneous information added to the knowledge base. The difference between the two tasks is captured by the verification task. In this paper our main goals are identification and verification, but we train our approach on information extraction, relying on the assumption that most claims made in the texts retrieved via the web search engine are accurate.

In related work, Nakashole and Mitchell (2014) developed an approach to verify subject-verb-object triples against a knowledge base, taking into account the objectivity of the language used in the sources stating the triple. Our approach is agnostic to the syntactic form of the claims, thus it can identify claims expressed in greater linguistic variety. Ciampaglia et al. (2015) fact-checked subject-predicate-object triples against a knowledge graph constructed from DBpedia, but they considered only the paths between the subject and the predicate in their algorithm thus ignoring the predicate itself. Dong et al. (2015) established the trustworthiness of a web source by comparing the subject-predicate-object triples extracted from it to the Knowledge Vault built by Google, but did not focus on claim identification and verification. Adar et al. (2009) developed an approach to detect inconsistencies between versions of Wikipedia in different languages, but they focused on manually extracted infoboxes. Finally, Vlachos and Riedel (2014) compiled a dataset of claims fact-checked by journalists, but the claims are much more complex than the ones we considered in this paper.

Other work that discussed the extraction of statistical properties includes the approaches of Hoffmann et al. (2010) and Intxaurrondo et al. (2015), both employing approximate matching to deal with the approximation of numerical values in text. In order to learn their model, Hoffmann et al. (2010) take advantage of the structure of the articles in Wikipedia developing a classifier that identifies the schema followed by each article, which is not straightforward to extend to texts beyond this source. Intxaurrondo et al. (2015) on the other hand focus on tweets and make the assumption that the entity discussed in each tweet is determined in advance, thus the extractor needs only to associate a numerical value with the property of interest, i.e. the task is reduced from triple extraction to labeling values.

7 Conclusions - Future work

In this paper we developed a distantly supervised approach for identification and verification of simple statistical claims. We evaluated both as statistical property extractor and as a claim identifier on 16 relations from Freebase. In future work we aim to improve our approach by taking into account continuous representations of the words in the patterns and to extend it to more complex claims, e.g. claims about change in financial indicators.

[2] Accessed in August 2015.

References

Eytan Adar, Michael Skinner, and Daniel S. Weld. 2009. Information arbitrage across multi-lingual wikipedia. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, pages 94–103.

Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M. Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. 2015. Computational fact checking from knowledge networks. *PLoS ONE*, 10(6):e0128193, 06.

Sarah Cohen, Chengkai Li, Jun Yang, and Cong Yu. 2011. Computational journalism: A call to arms to database researchers. In *Proceedings of the Conference on Innovative Data Systems Research*, volume 2011, pages 148–151.

Mark Craven and Johan Kumlien. 1999. Constructing biological knowledge-bases by extracting information from text sources. In *Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology*, pages 77–86.

Xin Luna Dong, Evgeniy Gabrilovich, Kevin Murphy, Van Dang, Wilko Horn, Camillo Lugaresi, Shaohua Sun, and Wei Zhang. 2015. Knowledge-based trust: Estimating the trustworthiness of web sources. pages 938–949.

Terry Flew, Anna Daniel, and Christina L. Spurgeon. 2012. The promise of computational journalism. In *Proceedings of the Australian and New Zealand Communication Association Conference*, pages 1–19.

Raphael Hoffmann, Congle Zhang, and Daniel S. Weld. 2010. Learning 5000 relational extractors. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, page 286295.

Rob J. Hyndman and Anne B. Koehler. 2006. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679 – 688.

Ander Intxaurrondo, Eneko Agirre, Oier Lopez de Lacalle, and Mihai Surdeanu. 2015. Diamonds in the rough: Event extraction from imperfect microblog data. In *Proceedings of the 2015 Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*.

Christian Kohlschütter, Peter Fankhauser, and Wolfgang Nejdl. 2010. Boilerplate detection using shallow text features. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, pages 441–450.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

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*, pages 1003–1011.

Ndapandula Nakashole and Tom M Mitchell. 2014. Languageaware truth assessment of fact candidates. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*.

Andreas Vlachos and Sebastian Riedel. 2014. Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technology and Computational Social Science*, July.