Entity Linking at Web Scale

Thomas Lin, Mausam, Oren Etzioni
Computer Science & Engineering
University of Washington
Seattle, WA 98195, USA
{tlin, mausam, etzioni}@cs.washington.edu

Abstract

This paper investigates entity linking over millions of high-precision extractions from a corpus of 500 million Web documents, toward the goal of creating a useful knowledge base of general facts. This paper is the first to report on entity linking over this many extractions, and describes new opportunities (such as corpus-level features) and challenges we found when entity linking at Web scale. We present several techniques that we developed and also lessons that we learned. We envision a future where information extraction and entity linking are paired to automatically generate knowledge bases with billions of assertions over millions of linked entities.

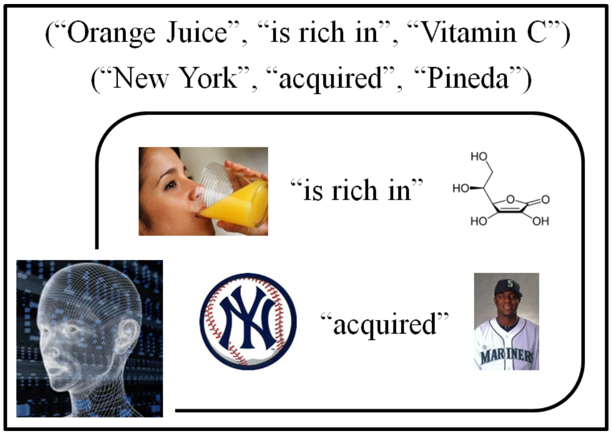
("Orange Juice", "is rich in", "Vitamin C")
("New York", "acquired", "Pineda")
"is rich in"
HO
HO
O
O
HO
OH
"acquired"
MARINERS

Figure 1: Entity Linking elevates textual argument strings to meaningful *entities* that hold properties, semantic types, and relationships with each other.

1 Introduction

Information Extraction techniques such as Open IE (Banko et al., 2007; Weld et al., 2008) operate at unprecedented scale. The ReVerb extractor (Fader et al., 2011) was run on 500 million Web pages, and extracted 6 billion ($Subject, Relation, Object$) extractions such as ("Orange Juice", "is rich in", "Vitamin C"), over millions of textual relations. Linking each textual argument string to its corresponding Wikipedia entity, known as *entity linking* (Bunescu and Paşca, 2006; Cucerzan, 2007), would offer benefits such as semantic type information, integration with linked data resources (Bizer et al., 2009), and disambiguation (see Figure 1).

Existing entity linking research has focused primarily on linking all the entities within individual documents into Wikipedia (Milne and Witten, 2008; Kulkarni et al., 2009; Dredze et al., 2010). To link a million documents they would repeat a million times. However, there are opportunities to do better when we know ahead of time that the task is large scale linking. For example, information on one document might help link an entity on another document. This relates to cross-document coreference (Singh et al., 2011), but is not the same because cross-document coreference does not offer all the benefits of linking to Wikipedia. Another opportunity is that after linking a million documents, we can discover systematic linking errors when particular entities are linked to many more times than expected.

In this paper we entity link millions of high-precision extractions from the Web, and present our initial methods for addressing some of the opportunities and practical challenges that arise when link-

Proc. of the Joint Workshop on Automatic Knowledge Base Construction & Web-scale Knowledge Extraction (AKBC-WEKEX), pages 84–88,
NAACL-HLT, Montréal, Canada, June 7-8, 2012. ©2012 Association for Computational Linguistics

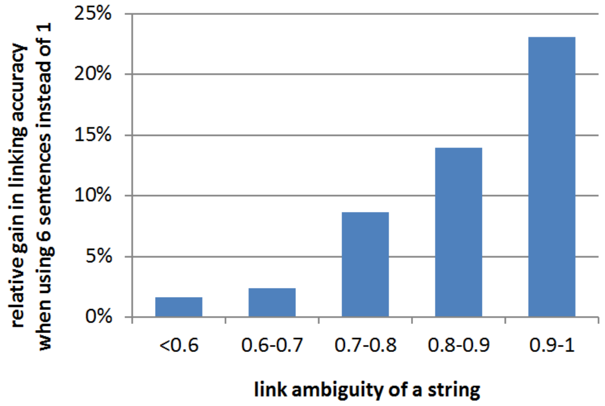

Figure 2: Context matching using more source sentences can increase entity linking accuracy, especially in cases where link ambiguity is high.

ing at this scale.

2 Entity Linking

Given a textual assertion, we aim to find the Wikipedia entity that corresponds to the argument. For example, the assertion ("New York", "acquired", "Pineda") should link to the Wikipedia article for *New York Yankees*, rather than *New York City*.

Speed is a practical concern when linking this many assertions, so instead of designing a system with sophisticated features that rely on the full Wikipedia graph structure, we instead start with a faster system leveraging linking features such as string matching, prominence, and context matching. (Ratinov et al., 2011) found that these "local" features already provide a baseline that is very difficult to beat with the more sophisticated "global" features that take more time to compute. For efficient high-quality entity linking of Web scale corpora, we focus on the faster techniques, and then later incorporate *corpus-level features* to increase precision.

2.1 Our Basic Linker

Given an entity string, we first obtain the most prominent Wikipedia entities that meet string matching criteria. As in (Fader et al., 2009), we measure prominence using inlink count, which is the number of Wikipedia pages that link to a Wikipedia entity's page. In our example, candidate matches for "New York" include entities such as:

- New York (State) at 92,253 inlinks
- New York City at 87,974 inlinks
- New York Yankees at 8,647 inlinks
- New York University at 7,983 inlinks

entity	assertions	wiki inlinks	ratio
"Barack Obama"	16,094	16,415	0.98
"Larry Page"	13,871	588	23.6
"Bill Clinton"	5,710	11,176	0.51
"Microsoft"	5,681	12,880	0.44
"Same"	6,975	36	193

Table 1: The ratio between an entity's *linked assertion count* and its *inlink prominence* can help to detect systematic errors to correct or filter out.

After obtaining a list of candidate entities, we employ a context matching (Bunescu and Paşca, 2006) step that uses cosine similarity to measure the semantic distance between the assertion and each candidate's Wikipedia article. For example, if our assertion came from the sentence "New York acquired Pineda on January 23," then we would calculate the similarity between this sentence and the Wikipedia articles for New York (State), New York City, etc.

As a final step we calculate a *link score* for each candidate as a product of string match level, prominence score and context match score. We also calculate a *link ambiguity* as the 2^{nd} highest link score divided by the highest link score. The best matches have high link score and low link ambiguity.

2.2 Corpus-Level Features

This section describes two novel features we used that are enabled when linking at the corpus level.

2.2.1 Collective Contexts

One corpus-level feature we leverage here is *collective contexts*. We observe that in an extraction corpus, the same assertion is often extracted from multiple source sentences across different documents. If we collect together the various source sentences, this can provide stronger context signal for entity linking. While "New York acquired Pineda on January 23" may not provide strong signal by itself, adding another source sentence such as "New York acquired Pineda to strengthen their pitching staff," could be enough for the linker to choose the *New York Yankees* over the others. Figure 2 shows the gain in linking accuracy we observed when using 6 randomly sampled source sentences per assertion instead of 1. At each *link ambiguity* level, we took 200 random samples.

2.2.2 Link Count Expectation

Another corpus-level feature we found to be useful is *link count expectation*. When linking millions of general assertions, we do not expect strong relative deviation between the *number of assertions linking to each entity* and the *known prominence of the entities*. For example, we would expect many more assertions to link to "Lady Gaga" than "Michael Pineda." We formalize this notion by calculating an *inlink ratio* for each entity as the *number of assertions linking to it* divided by its *inlink prominence*.

When linking 15 million assertions, we found that ratios significantly greater than 1 were often signs of systematic errors. Table 1 shows ratios for several entities that had many assertions linked to them. It turns out that many assertions of the form "(Page, loaded in, 0.23 seconds)" were being incorrectly linked to "Larry Page," and assertions like "(Same, goes for, women)" were being linked to a city in East Timor named "Same." We filtered systematic errors detected in this way, but these errors could also serve as valuable negative labels in training a better linker.

2.3 Speed and Accuracy

Some of the existing linking systems we looked at (Hoffart et al., 2011; Ratinov et al., 2011) can take up to several seconds to link documents, which makes them difficult to run at Web scale without massive distributed computing. By focusing on the fastest local features and then improving precision using corpus-level features, our initial implementation was able to link at an average speed of 60 assertions per second on a standard machine without using multithreading. This translated to 3 days to link the set of 15 million textual assertions that REVERB identified as having the highest precision over its run of 500 million Web pages. On our Figure 2 data, overall linking accuracy was above 70%.

2.4 Unlinkable Entities

Aside from the speed benefit, another advantage of using the "extract then entity-link" pipeline (rather than entity linking all the source documents and then running extraction only for linked entities) is that it also allows us to capture assertions concerning the long tail of entities (e.g., "prune juice") that are not prominent enough to have their own Wikipedia article. Wikipedia has dedicated articles for "apple juice" and "orange juice" but not for "prune juice" or "wheatgrass juice." Searching Wikipedia for "prune juice" instead redirects to the article for "prune," which works for an encyclopedia, but not for many entity linking end tasks because "prunes" and "prune juice" are not the same and do not have the same semantic types. Out of 15 million assertions, we observed that around 5 million could not be linked. Even if an argument is not in Wikipedia, we would still like to assign semantic types to it and disambiguate it. We have been exploring handling these *unlinkable entities* over 3 steps, which can be performed in sequence or jointly:

2.4.1 Detect Entities

Noun phrase extraction arguments that cannot be linked tend to be a mix of entities that are too new or not prominent enough (e.g., "prune juice," "fiscal year 2012") and non-entities (e.g., "such techniques," "serious change"). We have had success training a classifier to separate these categories by using features derived from the Google Books Ngrams corpus, as unlinkable entities tend to have different usage-over-time characteristics than non-entities. For example, many non-entities are seen in books usage going back hundreds of years, with little year-to-year frequency variation.

2.4.2 Predict Types

We found that we can predict the Freebase types of unlinkable entities using instance-to-instance class propagation from the linked entities. For example, if "prune juice" cannot be linked, we can predict its semantic types by observing that the collection of relations it appears with (e.g., "is a good source of") also occur with linkable entities such as "orange juice" and "apple juice," and propagate the semantic types from these similar entities. Linking at Web scale means that unlinkable entities often have many relations to use for this process. When available, shared term heads (e.g., "juice") could also serve as a signal for finding entities that are likely to share semantic types.

	top assertions
rank by freq	"(teachers, teach at, school)" "(friend, teaches at, school)" "(Mike, teaches at, school)" "(biologist, teaches at, Harvard)" "(Jorie Graham, teaches at, Harvard)"
rank by link score	"(Pauline Oliveros, teaches at, RPI)" "(Azar Nafisi, teaches at, Johns Hopkins)" "(Steven Plaut, teaches at, Univ of Haifa)" "(Niall Ferguson, teaches at, NYU)" "(Ha Jin, teaches at, Boston University)"

Table 2: Ranking based on *link score* gives higher quality results than ranking based on *frequency*.

2.4.3 Disambiguation

In cases where we predict mutually exclusive types (e.g., *film* and *person* can be observed to be mutually exclusive in Freebase instances), this signifies that the argument is a name shared by multiple entities. We plan to use clustering to recover the most likely types of the multiple entities and then divide assertions among them.

3 Resources Enabled

We observed that entity linking of 15 million textual extractions enables several resources.

3.1 Freebase Selectional Preferences

Each Wikipedia entity that gets linked is easily annoted with its Freebase (Bollacker et al., 2008) semantic types using data from the Freebase Wikipedia Extraction (WEX) project. On the 15 million extractions, the entities that we linked to encompassed over 1,300 Freebase types. Knowing these entity types then allows us to compute the Freebase selectional preferences of all our textual relations. For example, we can observe from our linked entities that the "originated in" relation most often has types such as *food*, *sport*, and *animal breed* in the domain. Selectional preferences have been calculated for WordNet (Agirre and Martinez, 2002), but have not been calculated at scale for Freebase, which is something that we get for free in our scenario. Freebase has a much greater focus on named entities than WordNet, so these selectional preferences could be valuable in future applications.

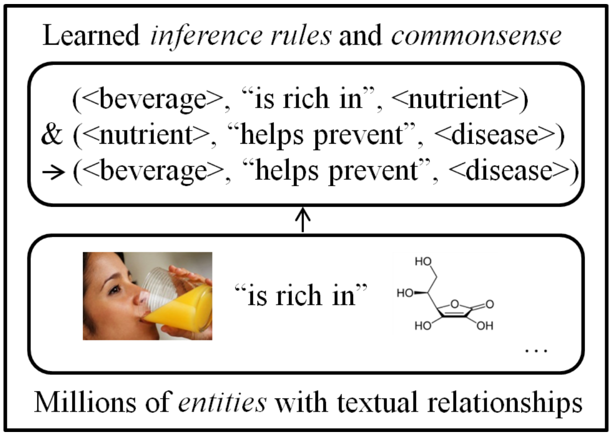
Learned *inference rules* and *commonsense*
(<beverage>, "is rich in", <nutrient>)
& (<nutrient>, "helps prevent", <disease>)
→ (<beverage>, "helps prevent", <disease>)
"is rich in"
HO
HO
O
O
HO
OH
…
Millions of *entities* with textual relationships

Figure 3: On top of an Entity Linked Open IE corpus we could learn *inference rules* and *commonsense knowledge*.

3.2 Improved Instance Ranking Function

We observed *link score* to be a better ranking function than *assertion frequency* for presenting query results. For example, Table 2 shows the top results when searching the extractions for instances of the "teaches at" textual relation. When results are sorted by frequency in the corpus, assertions like "(friend, teaches at, school)" and "(Mike, teaches at, school)" are returned first. When results are sorted by *link score*, the top hundred results are all specific instances of professors and the schools they teach at, and are noticeably more specific and generally correct than the top frequency-sorted instances.

3.3 Inference

Disambiguated and typed entities are especially valuable for inference applications over extracted data. For example if we observe enough instances like "Orange Juice is rich in Vitamin C," "Vitamin C helps prevent scurvy," and "Orange Juice helps prevent scurvy," then we can learn the inference rule shown in Figure 3. (Schoenmackers et al., 2010) explored this, but without entity linking they had to rely on heavy filtering against hypernym data, losing most of their extraction instances in the process. We plan to explore how much gain we get in inference rule learning when using entity linking instead of hypernym filtering. Linked instances would also be higher precision input than what is currently available for learning implicit common sense properties of textual relations (Lin et al., 2010).

4 Conclusions

While numerous entity-linking systems have been developed in recent years, we believe that going forward, researchers will increasingly be considering the opportunities and challenges that arise when scaling up from the single document level toward the Web-scale corpus level. This paper is the first to run and report back on entity linking over millions of textual extractions, and we proposed novel ideas in areas such as corpus-level features and unlinkable entities. There are potentially many other corpus-level features and characteristics to explore, as well as additional challenges (e.g., how to best evaluate recall at this scale), and we look forward to seeing additional research in Entity Linking at Web scale over the coming years.

5 Acknowledgements

This research was supported in part by NSF grant IIS-0803481, ONR grant N00014-08-1-0431, and DARPA contract FA8750-09-C-0179, and carried out at the University of Washington's Turing Center.

References

Eneko Agirre and David Martinez. 2002. Integrating selectional preferences in wordnet. In *Proceedings of the first International WordNet Conference*.

Michele Banko, Michael J Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the Web. In *Proceedings of IJCAI*.

Christian Bizer, Tom Heath, and Tim Berners-Lee. 2009. Linked data - the story so far. In *International Journal on Semantic Web and Information Systems (IJSWIS)*.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of SIGMOD '08*, pages 1247–1250, New York, NY, USA.

Razvan Bunescu and Marius Paşca. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association of Computational Linguistics (EACL)*.

Silviu Cucerzan. 2007. Large-scale named entity disambiguation based on wikipedia data. In *Proceedings of EMNLP*.

Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber, and Tim Finin. 2010. Entity disambiguation for knowledge base population. In *Proceedings of COLING*.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2009. Scaling Wikipedia-based named entity disambiguation to arbitrary Web text. In *IJCAI-09 Workshop on User-contributed Knowledge and Artificial Intelligence (WikiAI09)*.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of EMNLP*.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. 2011. Robust disambiguation of named entities in text. In *Proceedings of EMNLP*.

Sayali Kulkarni, Amit Singh, Ganesh Ramakrishnan, and Soumen Chakrabarti. 2009. Collective annotation of wikipedia entities in text. In *Proceedings of KDD*.

Thomas Lin, Mausam, and Oren Etzioni. 2010. Commonsense from the web: Relation properties. In *AAAI Fall Symposium on Commonsense*.

David Milne and Ian H. Witten. 2008. Learning to link with wikipedia. In *Proceedings of the 17h ACM International Conference on Information and Knowledge Management (CIKM)*.

Lev Ratinov, Dan Roth, Doug Downey, and Mike Anderson. 2011. Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (ACL)*.

Stefan Schoenmackers, Oren Etzioni, Daniel S. Weld, and Jesse Davis. 2010. Learning first-order horn clauses from web text. In *Proceedings of EMNLP*.

Sameer Singh, Amarnag Subramanya, Fernando Pereira, and Andrew McCallum. 2011. Large-scale cross-document coreference using distributed inference and hierarchical models. In *Proceedings of ACL*.

Daniel S. Weld, Raphael Hoffmann, and Fei Wu. 2008. Using wikipedia to bootstrap open information extraction. In *SIGMOD Record*.