Using Natural Language to Integrate, Evaluate, and Optimize Extracted Knowledge Bases

Doug Downey, Chandra Sekhar Bhagavatula
Dept. of Elec. Eng. & Comp. Sci., Northwestern University, Evanston IL
ddowney@eecs.northwestern.edu,csbhagav@u.northwestern.edu

ABSTRACT

Web Information Extraction (WIE) systems can extract billions of unique facts, but integrating the assertions into a coherent knowledge base and evaluating across different WIE techniques remains a challenge. We propose a framework that utilizes natural language to integrate and evaluate extracted knowledge bases (KBs). In the framework, KBs are integrated by exchanging probability distributions over natural language, and evaluated by how well the output distributions predict held-out text. We describe the advantages of the approach, and detail remaining research challenges.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information search and retrieval; H.3.5 [**Information Storage and Retrieval**]: Online Information ServicesWeb-based services

General Terms

Algorithms, Measurement, Experimentation

1. INTRODUCTION

Extracting knowledge automatically from the Web is known as Web Information Extraction (WIE), and is a task of broad and increasing interest. Over the past decade, a variety of research studies and prototypes have investigated WIE techniques [1–11]. WIE has recently been pursued in industry in the form of question-answering systems like IBM's Watson [12] and Web search aids such as the Google Knowledge Graph and Microsoft Satori. WIE presents a promising route toward achieving Tim Berners-Lee's vision of a Semantic Web, and one day acquiring the knowledge required to enable human-level artificial intelligence.

Existing WIE systems vary along two key dimensions: the *type of content* they target for extraction (Web tables, text, Wikipedia, etc.), and the *representation of the extracted knowledge* (individual tuples or frames, or additions to given ontology). Because WIE systems are so diverse, it is difficult to integrate knowledge across extracted KBs, or to evaluate across distinct extraction approaches. To deliver on the promise of WIE, new methods are required to allow different system builders to work together to construct a massive body of knowledge.

In this position paper, we propose a framework for integrating and evaluating WIE systems. The approach hinges on representing extracted knowledge in terms of *probability distributions over natural language (NL)*. Many existing WIE systems already utilize such distributions as input, at least implicitly—as a simple example, the distribution of terms C and x in the extraction pattern "C such as x" is commonly used to extract x's that are members of the class C, as in the phrase "cities such as Boston" [2, 13]. Assertions that occur more frequently in text (i.e., for which the extraction pattern has higher probability) are deemed more likely to be correct [14]. Our contention in this paper is that a generalization of this capability, in which KBs import and export distributions over language, can enable automated integration of WIE systems. Further, we believe that the quality of the output distributions (according to some measure) forms a promising metric for evaluating and optimizing WIE systems.

We envision a large-scale research effort in which different parties continuously extract KBs in a variety of ways, and the KBs selectively share knowledge with each other in natural language in an effort to encode a vast, high-precision, globally-interoperable body of knowledge. As discussed in Sections 3 and 4 below, utilizing NL for knowledge base integration and evaluation has distinct advantages: it enables KB integration without requiring a commitment to any single ontology, and it enables KB optimization over trillions of readily available evaluation examples (in the form of running text on the Web). However, operationalizing the proposed approach entails a number of research challenges, detailed in Section 5. We begin by discussing previous work in WIE.

2. WEB INFORMATION EXTRACTION

Web Information Extraction (WIE) is the task of extracting knowledge from content on the Web. Different WIE approaches target different types of Web content. Some systems extract knowledge from text across the Web [15], while others focus on Wikipedia text or infoboxes [16, 17] or Web tables [5, 6]. Other approaches integrate knowledge solicited from Web contributors [18]. Together, these KBs contain billions of facts, spanning an enormous variety of topics.

WIE approaches also differ significantly in the degree of representational structure in the extracted knowledge. For example, some approaches extract independent propositions or tuples (e.g. `MayorOf(New York City, Bloomberg)`) [4, 19], while others extract more comprehensive semantic frames [11]. Some approaches organize extracted facts into an ontology: these range from lightweight ontologies, often rooted in Wikipedia [7–10], to rich knowledge representation systems such as Cyc [1, 3].

The diversity of extracted facts and knowledge representation schemes presents a significant difficulty: it is unclear how to best combine different systems or evaluate across different systems. We present our proposed solution to these problems in the following sections.

3. NL FOR INTEGRATING KBS

If two KBs contain different sets of knowledge, represented in different ways, how can the KBs share knowledge with each other?

Previous work on this task includes *data integration* approaches from the database community, which attempt to merge two different KBs into a single knowledge base [20]. This approach is limited in that it generally requires special-purpose engineering or training examples for each *pair* of KBs to be integrated.

A distinct, potentially more scalable approach involves choosing one or more common reference ontologies with which all other KBs can be integrated. This approach is employed in the Linked Open Data project [21], in which different knowledge bases link their statements to a handful of common shared vocabularies. Wikipedia is a common reference ontology for this task, and semi-automated methods for integrating with Wikipedia have been proposed for Cyc [22], relational databases [23], and tuples extracted from text [24]. This approach, while potentially much more scalable than pairwise integration, is also heavily restrictive: a small number of reference knowledge bases must be selected, and choosing such KBs is difficult. Further, changes to a reference knowledge base can entail burdensome updates to how each KB exports knowledge. While we believe integration with Linked Open Data is an important component of KB integration (and we discuss how to incorporate it in our approach in Section 5), we believe natural language integration has distinct advantages as discussed below.

3.1 The NL Protocol

We propose an approach in which knowledge bases are integrated by exchanging natural language. Because KBs will typically have uncertainty associated with their knowledge and how it is expressed in language, we define a protocol that exchanges not raw text but instead probability distributions over language.

Formally, the **NL protocol** requires that each KB be capable of two operations. First, we require that a given KB K can update its knowledge (which we denote as a change in knowledge ΔK) based on a probability distribution over *sentences* $\mathbf{w}$ of natural language:

$$P(\mathbf{w}) \rightarrow \Delta K \tag{1}$$

Secondly, we require the inverse operation, that a knowledge base be capable of exporting its knowledge in terms of a probability distribution over sentences:

$$K \rightarrow P_K(\mathbf{w}) \tag{2}$$

We propose that KBs exchange knowledge by, at an abstract level, reading and writing distributions $P(\mathbf{w})$. In practice, knowledge bases may choose not to output complete distributions, but instead exchange certain agreed-upon *properties* of the distributions. A concrete example is given in Section 3.2 below.

The primary advantages of the NL protocol are three-fold. First, the communication medium (natural language) is extremely **expressive**, and not tied to any single ontology. Each KB must implement two methods, for reading and writing knowledge in natural language, and it can then integrate with every other KB implementing the protocol. Secondly, knowledge exchanged in the NL protocol is **durable**: a fixed distribution $P_K(\mathbf{w})$ output by K at some point in time remains informative, even if K later changes radically. Third, the protocol is readily **interpretable** by humans, because it is expressed in natural language.

An immediate concern is whether producing the mappings in Equations 1 and 2 is feasible in practice. While fully exploiting the protocol requires overcoming a number of difficult research challenges (see Section 5), we are optimistic that the approach *is* practical. First, note that Equation 1 essentially formalizes a capability held *already* by many WIE systems. In particular, many free text extractors perform extraction from a corpus of distinct sentences S (e.g., [2, 4]). If we view the corpus in terms of a simple maximum-likelihood language model that assigns probability $P(\mathbf{w}) = 1/|S|$ for $\mathbf{w} \in S$ and zero otherwise, the existing extraction mechanisms supply exactly the mapping required in Equation 1.[1]

The natural language generation task, Equation 2, is less well-studied. However, below we illustrate how even KBs that do not use language for extraction, like those that perform extraction from Web tables, can provide valuable information about $P(\mathbf{w})$ through straightforward use of the *distributional hypothesis*, the notion that terms with similar meanings tend to appear in similar contexts [25].

3.2 Concrete Example

Some IE approaches that use language models as a foundation (e.g. [26]) could directly output a relatively compact, complete distribution $P(\mathbf{w})$. However, for most KBs, exporting an entire knowledge base—i.e., outputting a complete model $P(\mathbf{w})$ through a complex language generation process—may be undesirable. It may instead be preferable for KBs to periodically *query* each other about agreed-upon *properties* of $P(\mathbf{w})$, rather than exchanging complete language models. The following example illustrates how this might work, and also illustrates how the protocol can be employed even by WIE systems that do not extract knowledge from free text.

Consider a system that wishes to populate an initially-empty knowledge base K with a list of cities containing skyscrapers. Assume the system has a set of lexical extraction patterns (as in e.g. [2]), and thus aims to obtain strings x that yield high values of the product:

$$P(x \text{ and other cities}) * P(\text{skyscraper in } x) \tag{3}$$

Since K lacks this knowledge, it could pose a *language model (LM) query* to another KB K'. LM queries request

[1] Here, knowledge is derived from sentences considered independently. Relaxing this criterion to allow for richer context beyond a single sentence is an important direction, discussed in Section 5.

products of probabilities of strings, where the strings are composed of surface terms and one or more free variables (x is a free variable in the query in Equation 3). The response to a LM query is a list of x's and the estimated value of the product for each.

Assume that K' is populated by extraction not from free text, but instead from Wikipedia tables. K' lacks the textual evidence to service the LM query itself, so it might pose the same query to another text-based extractor K''. K'' extracts from a textual corpus that includes three answers with positive probability: Shanghai, New York City, and Montreal. For concreteness, assume K'' simply returns a distribution where each of the three cities has probability 0.2, and the remaining 0.4 of probability mass is distributed uniformly over other phrases.

Based on the three "seed" examples with non-negligible probability, the table extractor K' might attempt to impute a more accurate distribution over x using its KB of tables along with the distributional hypothesis of language. Specifically, given a table column that contains the seed cities, the other cells S in the same column are likely to be semantically similar to the seeds. Thus, K' can adjust the distribution it returns to give higher probability to the strings S, and thereby return a more accurate distribution.

In the example, the first table returned by the Wikitables extraction system [27] when queried for the seeds lists the top 40 cities ranked in terms of "Global City Competitiveness Index." While this table does not explicitly refer to skyscrapers, it happens that all 40 listed cities do, in fact, contain skyscrapers. Thus, when K' adjusts its query response to return positive probability for these 40 cities, K will receive a response that has perfect precision and dramatically higher recall than the original three strings returned by K''.

This example illustrates how even when a knowledge base may not contain the target relation or be based on text (e.g. K' need not contain any table listing all cities with skyscrapers), it can leverage other KBs (in this case K'') in order to answer queries. By exchanging knowledge, the three KBs in the example are able to produce an answer to a query that is dramatically better than any of the KBs could produce in isolation.

4. NL FOR EVALUATING AND OPTIMIZING KBS

How to estimate the quality of an extracted KB is an open question. Previous work has emphasized that extracted KBs must ultimately be evaluated in terms of "end-tasks," such as decision making and question answering [28]. While end task evaluation is necessary to ensure the knowledge can yield useful technology, it can be cost-prohibitive: evaluation with end-tasks generally requires direct human judgments of performance. Unless we solicit new human input often, we risk overfitting to a fixed objective. Thus, while we believe end task evaluations are vital to periodically evaluate competing approaches, due to their cost they cannot be used for continuous optimization of KB constructors.

4.1 The NL Objective

We propose the **NL Objective** for KBs, in which a KB K is evaluated in terms of how accurately its output distribution $P_K(\mathbf{w})$ predicts held-out text. This objective has the advantage of not being biased toward a limited data set; in fact, *trillions* of examples for training and evaluation are available, in the form of running text on the Web.

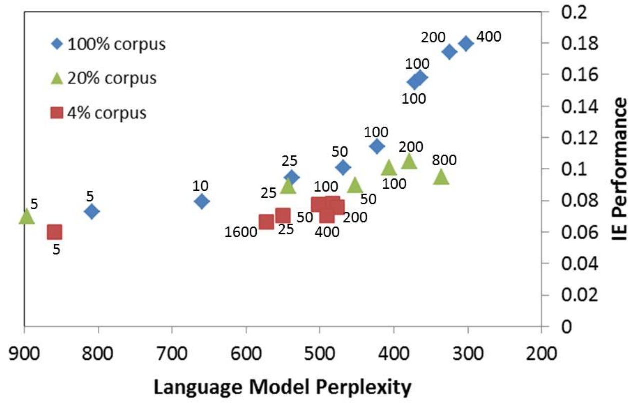

Figure 1: Web Information Extraction (WIE) performance of a Hidden Markov Model, as accuracy of the HMM's language model $P(\mathbf{w})$ varies (reprinted from [29]). Number labels indicate the number of latent states in the HMM, and performance is shown for three training corpus sizes (the full corpus consists of approximately 60 million tokens). WIE accuracy (in terms of area under the precision-recall curve) tends to increase as language modeling accuracy improves (i.e. perplexity decreases). WIE accuracy correlates more strongly with perplexity (-0.88, Spearman's) than with corpus size (0.71) or number of latent states (0.38).

How to measure the "accuracy" of P_K is an important question. Fortunately, there is some indication that other typical measures of extraction performance, like precision and recall of extracted relations, correlates with the standard perplexity metric used in language modeling. In Figure 1, we show experiments from previous work [29] that demonstrate how the perplexity of a Hidden Markov Model correlates strongly with the model's accuracy in a standard "set expansion" WIE task.

5. KEY CHALLENGES

The NL Protocol and NL Objective present advantages as well as unique challenges. In this section, we detail key limitations of the techniques and remaining research challenges.

5.1 Developing APIs in the NL Protocol

A primary limitation of the NL Protocol is that—like natural language, but unlike Linked Open Data RDF—messages in the NL Protocol are ambiguous. Given only the statement that "Chicago is a city," it is unclear whether the string Chicago refers to a fictional city or a real one, or whether it is the same meaning of Chicago in the phrase "that song by Chicago was playing on the radio."

We believe the ambiguity of language can be overcome to build a powerful protocol, through the use of well-designed queries. The "language model query" in the example in Section 3, for example, mitigates ambiguity by querying for a product of two distinct language probabilities. It is less likely that an erroneous or ambiguous phrase occurs in *both* the skyscraper and the city context, when compared to a single context alone. By composing larger products of additional indicative phrases (e.g. "cities including x," "x and other cities," etc.) we would expect that, as the number of phrases increases and if probability estimates are accurate, the high-probability x's would correspond to correct answers for any query. In fact, it can be proved that under assumptions that hold approximately in large corpora, such extraction techniques are *guaranteed* to achieve high accuracy [14].

To make sense disambiguation explicit, it would be possible to augment the NL protocol to allow not only surface strings, but also word classes of various types (e.g. parts of speech or well-established semantic classes). One option is to allow terms in queries that are not surface strings, but instead indicate a reference to a particular Linked Data URI, e.g., "x is the mayor of <reference to en.wikipedia.org/Chicago, Illinois>." This would allow KBs to leverage Linked Open Data URIs where they are well-established, but back off to natural language in other cases. Of course, utilizing URIs sacrifices some of the advantages of the NL protocol discussed above. New methods are needed to determine when utilizing URIs rather than "pure" NL is appropriate.

Lastly, while in the NL protocol we focus on single sentences $\mathbf{w}$, the API can be generalized to a richer discourse model. One simple example would allow distributions $P(\mathbf{w}|\mathbf{t})$, where the distribution over sentences is conditioned on a given vector of terms $\mathbf{t}$ appearing in the *document* containing the sentence. We note that given sufficient contextual information, word senses may become unambiguous from the context, obviating some of the need for non-NL URIs.

5.2 A Market for Knowledge

We expect that different KBs, constructed using different methods and for different purposes, will specialize in different knowledge. As a result, a new category of services need to be developed that can advise a KB about which other KBs can answer which questions. How should a KB combine evidence across other KBs? And what incentive structures will reward KBs for providing high-quality output, and help WIE systems focus on extracting new knowledge that is helpful to other KBs? New mechanisms would need to be designed for these purposes.

5.3 Clarifying the LM Objective

While the experiments in Section 4 show that the perplexity measure correlates well with IE performance for a particular class of models (HMMs), in general the LM Objective requires further refinement.

In the limit, a KB that performs sufficiently well according to the LM Objective will necessarily contain a vast, useful body of knowledge. However, in the nearer-term, optimizing the LM Objective in terms of standard metrics like perplexity has the potential to be counter-productive. As a specific example, extractors based on trigram models can actually be shown to be *less* accurate for WIE than HMMs, but the trigrams achieve better perplexity scores in modeling $P(\mathbf{w})$ through the use of careful backoff and smoothing techniques.

Thus, new metrics must be developed for evaluating distributions $P(\mathbf{w})$. We desire metrics that vary monotonically with the "knowledge content" of a KB. The ideal metric would penalize *semantic* errors, rather than (for example) rewarding particularly precise probability estimates on common phrases. The recent "adversarial evaluation" approach for NL models represents one promising direction [30].

It is also important to note that even with trillions of training examples, language model performance will typically remain an imperfect measure of KB capabilities. For example, the LM Objective may never completely capture whether a KB can perform arithmetic (plenty of reasonable sums, such as "25 plus 329," never occur even in a corpus as large as the Web). Characterizing what types of knowledge are, and are not, reflected by the NL Objective is an important task.

5.4 Applications

Systems that can utilize KBs for end tasks would help guide the acquisition of new knowledge toward useful ends. While many extraction techniques have been developed, relatively less effort has been spent on developing applications that allow users to interactively search, browse, and mine the extracted knowledge. Augmenting existing Web search results with extracted knowledge (as in the Google Knowledge Graph and Microsoft's Satori) is one possible direction. Other examples include search tools for extracted information, such as TextRunner [4] and Google Fusion Tables [31], and extraction visualization tools like Atlasify [32]. Massive extracted KBs present exciting opportunities to develop compelling new end-user applications.

5.5 Scaling LM training

The LM Objective suggests that building more predictive language models is an important direction for WIE. In particular, we require language models that can handle potentially lengthy queries—i.e. models that infer which strings are likely to occur, even if the strings never appear on the Web. Latent variable models such as HMMs are one step toward such a model; other promising avenues include deep neural network language models [33] and recent models that include language compositionality [34, 35].

While parallel training techniques have been developed for many models [36, 37], training sophisticated models on large corpora with large vocabularies is an ongoing challenge. Can we develop new techniques that actively select *human* input to improve the models? In some settings, carefully selecting informative input can dramatically reduce the amount of training required [38], but these techniques have not been applied to modern statistical language models. A related direction involves developing new learning approaches that do not iterate over the entire corpus, but instead learn from selected *statistics* computed over the data (e.g. [39]).

6. CONCLUSION

We proposed a framework that utilizes natural language for integrating, evaluating, and optimizing extracted knowledge bases (KBs). In the NL protocol, KBs exchange knowledge by asking and answering queries about probability distributions over language. The NL Objective evaluates and optimizes KBs in terms of their ability to accurately estimate probabilities over language. Several research challenges remain. Our next steps include implementing the NL protocol over existing extracted KBs, and evaluating its effectiveness experimentally.

References

[1] Cynthia Matuszek Michael, Michael Witbrock, Robert C. Kahlert, John Cabral, Dave Schneider, Purvesh Shah, and Doug Lenat. Searching for common sense: Populating cyc from the web. In *In Proceedings of the Twentieth National Conference on Artificial Intelligence*, pages 1430–1435, 2005.

[2] O. Etzioni, M. Cafarella, D. Downey, S. Kok, A. Popescu, T. Shaked, S. Soderland, D. Weld, and A. Yates. Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence*, 165(1):91–134, 2005.

[3] Kenneth D Forbus, Christopher Riesbeck, Lawrence Birnbaum, Kevin Livingston, Abhishek Sharma, and Leo Ureel. Integrating natural language, knowledge representation and reasoning, and analogical processing to learn by reading. In *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, volume 22, page 1542. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

[4] M. Banko, M. Cafarella, S. Soderland, M. Broadhead, and O. Etzioni. Open information extraction from the Web. In *Procs. of IJCAI*, 2007.

[5] Wolfgang Gatterbauer, Paul Bohunsky, Marcus Herzog, Bernhard Krüpl, and Bernhard Pollak. Towards domain-independent information extraction from web tables. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 71–80, New York, NY, USA, 2007. ACM.

[6] Michael J. Cafarella, Alon Y. Halevy, Daisy Z. Wang, Eugene W. 0002, and Yang Zhang. Webtables: exploring the power of tables on the web. *PVLDB*, 1(1):538–549, 2008.

[7] Fei Wu and Daniel S. Weld. Automatically refining the wikipedia infobox ontology. In *Proc. of WWW*, 2008.

[8] F.M. Suchanek, G. Kasneci, and G. Weikum. Yago: A core of semantic knowledge. In *Procs. of WWW*, 2007.

[9] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010)*, 2010.

[10] Sören Auer and Jens Lehmann. What have innsbruck and leipzig in common? extracting semantics from wiki content. In *Proc. of ESWC*, 2007.

[11] James Fan, David Ferrucci, David Gondek, and Aditya Kalyanpur. Prismatic: Inducing knowledge from a large scale lexicalized relation resource. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, pages 122–127. Association for Computational Linguistics, 2010.

[12] David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A Kalyanpur, Adam Lally, J William Murdock, Eric Nyberg, John Prager, et al. Building watson: An overview of the deepqa project. *AI magazine*, 31(3):59–79, 2010.

[13] M. Hearst. Automatic Acquisition of Hyponyms from Large Text Corpora. In *Procs. of the 14th International Conference on Computational Linguistics*, pages 539–545, Nantes, France, 1992.

[14] Doug Downey, Oren Etzioni, and Stephen Soderland. Analysis of a probabilistic model of redundancy in unsupervised information extraction. *Artificial Intelligence*, 174(11):726 – 748, 2010.

[15] Marius Pasca, Dekang Lin, Jeffrey Bigham, Andrei Lifchits, and Alpa Jain. Organizing and searching the world wide web of facts - step one: The one-million fact extraction challenge. In *AAAI 2006*. AAAI Press, 2006.

[16] Fei Wu, Raphael Hoffmann, and Daniel S. Weld. Information extraction from wikipedia: moving down the long tail. In *Proc. of KDD*, 2008.

[17] Fei Wu and Daniel S. Weld. Autonomously semantifying wikipedia. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 41–50, New York, NY, USA, 2007. ACM.

[18] Push Singh, Thomas Lin, Erik T Mueller, Grace Lim, Travell Perkins, and Wan Li Zhu. Open mind common sense: Knowledge acquisition from the general public. In *On the Move to Meaningful Internet Systems 2002: CoopIS, DOA, and ODBASE*, pages 1223–1237. Springer, 2002.

[19] L.K. Schubert and M.H. Tong. Extracting and evaluating general world knowledge from the brown corpus. In *Proc. of the HLT/NAACL Workshop on Text Meaning*, 2003.

[20] AnHai Doan and Alon Y. Halevy. Semantic-integration research in the database community. *AI Mag.*, 26(1):83–94, 2005.

[21] Christian Bizer, Tom Heath, Kingsley Idehen, and Tim Berners-Lee. Linked data on the web (ldow2008). In *Proceedings of the 17th international conference on World Wide Web*, pages 1265–1266. ACM, 2008.

[22] O. Medelyan and C. Legg. Integrating cyc and wikipedia: Folksonomy meets rigorously defined common-sense. In *Proc. of WIKIAI*, 2008.

[23] D. Downey, A. Ahuja, and M. Anderson. Learning to integrate relational databases with wikipedia. In *Proc. of WIKIAI*, 2009.

[24] Thomas Lin, Oren Etzioni, et al. Entity linking at web scale. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, pages 84–88. Association for Computational Linguistics, 2012.

[25] Z. Harris. Distributional structure. In J. J. Katz, editor, *The Philosophy of Linguistics*, pages 26–47. New York: Oxford University Press, 1985.

[26] Arun Ahuja and Doug Downey. Improved extraction assessment through better language models. In *Human Language Technologies: Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT)*, 2010.

[27] Chandra Sekhar Bhagavatula, Thanapon Noraset, and Doug Downey. Methods for Exploring and Mining Tables on Wikipedia. In *Proceedings of the ACM SIGKDD Interactive Data Exploration and Analytics (IDEA)*. ACM, 2013.

[28] Hoifung Poon, Janara Christensen, Pedro Domingos, Oren Etzioni, Raphael Hoffmann, Chloe Kiddon, Thomas Lin, Xiao Ling, Alan Ritter, Stefan Schoenmackers, et al. Machine reading at the university of washington. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, pages 87–95. Association for Computational Linguistics, 2010.

[29] Fei Huang, Arun Ahuja, Doug Downey, Yi Yang, Yuhong Guo, and Alexander Yates. Learning Representations for Weakly Supervised Natural Language Processing Tasks. *Computational Linguistics*, xx:yy, 2013.

[30] Noah A Smith. Adversarial evaluation for models of natural language. *arXiv preprint arXiv:1207.0245*, 2012.

[31] Hector Gonzalez, Alon Y Halevy, Christian S Jensen, Anno Langen, Jayant Madhavan, Rebecca Shapley, Warren Shen, and Jonathan Goldberg-Kidon. Google fusion tables: web-centered data management and collaboration. In *Proceedings of the 2010 international conference on Management of data*, pages 1061–1066. ACM, 2010.

[32] Brent Hecht, Samuel H Carton, Mahmood Quaderi, Johannes Schöning, Martin Raubal, Darren Gergle, and Doug Downey. Explanatory semantic relatedness and explicit spatialization for exploratory search. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 415–424. ACM, 2012.

[33] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008.

[34] Jeff Mitchell and Mirella Lapata. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1429, 2010.

[35] Richard Socher, Brody Huval, Christopher D Manning, and Andrew Y Ng. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1201–1211. Association for Computational Linguistics, 2012.

[36] Jason Wolfe, Aria Haghighi, and Dan Klein. Fully distributed em for very large datasets. In *ICML*, 2008.

[37] Yi Yang, Alexander Yates, and Doug Downey. Overcoming the memory bottleneck in distributed training of latent variable models of text. In *Proceedings of NAACL-HLT*, pages 579–584, 2013.

[38] Burr Settles. Active learning literature survey. *University of Wisconsin, Madison*, 2010.

[39] Michael Lucas and Doug Downey. Scaling semi-supervised naive bayes with feature marginals. In *Proceedings of ACL*, 2013.