Improved Extraction Assessment through Better Language Models

Arun Ahuja, Doug Downey
EECS Dept., Northwestern University
Evanston, IL 60208
{arun.ahuja, ddowney}@eecs.northwestern.edu

Abstract

A variety of information extraction techniques rely on the fact that instances of the same relation are "distributionally similar," in that they tend to appear in similar textual contexts. We demonstrate that extraction accuracy depends heavily on the accuracy of the language model utilized to estimate distributional similarity. An unsupervised model selection technique based on this observation is shown to reduce extraction and type-checking error by 26% over previous results, in experiments with Hidden Markov Models. The results suggest that optimizing statistical language models over unlabeled data is a promising direction for improving weakly supervised and unsupervised information extraction.

1 Introduction

Many weakly supervised and unsupervised information extraction techniques assess the correctness of extractions using the *distributional hypothesis*—the notion that words with similar meanings tend to occur in similar contexts (Harris, 1985). A candidate extraction of a relation is deemed more likely to be correct when it appears in contexts similar to those of "seed" instances of the relation, where the seeds may be specified by hand (Paşca et al., 2006), taken from an existing, incomplete knowledge base (Snow et al., 2006; Pantel et al., 2009), or obtained in an unsupervised manner using a generic extractor (Banko et al., 2007). We refer to this technique as *Assessment by Distributional Similarity* (ADS).

Typically, distributional similarity is computed by comparing co-occurrence counts of extractions and seeds with various contexts found in the corpus. *Statistical Language Models* (SLMs) include methods for more accurately estimating co-occurrence probabilities via back-off, smoothing, and clustering techniques (e.g. (Chen and Goodman, 1996; Rabiner, 1989; Bell et al., 1990)). Because SLMs can be trained from only unlabeled text, they can be applied for ADS even when the relations of interest are not specified in advance (Downey et al., 2007). Unlabeled text is abundant in large corpora like the Web, making nearly-ceaseless automated optimization of SLMs possible. But how fruitful is such an effort likely to be—to what extent does optimizing a language model over a fixed corpus lead to improvements in assessment accuracy?

In this paper, we show that an ADS technique based on SLMs is improved substantially when the language model it employs becomes more accurate. In a large-scale set of experiments, we quantify how language model perplexity correlates with ADS performance over multiple data sets and SLM techniques. The experiments show that accuracy over unlabeled data can be used for selecting among SLMs—for an ADS approach utilizing Hidden Markov Models, this results in an average error reduction of 26% over previous results in extraction and type-checking tasks.

2 Extraction Assessment with Language Models

We begin by formally defining the extraction and typechecking tasks we consider, then discuss statistical language models and their utilization for extraction assessment.

Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the ACL, pages 225–228,
Los Angeles, California, June 2010. ©2010 Association for Computational Linguistics

The extraction task we consider is formalized as follows: given a corpus, a target relation R, a list of seed instances S_R, and a list of candidate extractions U_R, the task is to order elements of U_R such that correct instances for R are ranked above extraction errors. Let U_{Ri} denote the set of the ith arguments of the extractions in U_R, and let S_{Ri} be defined similarly for the seed set S_R. For relations of arity greater than one, we consider the *typechecking* task, an important sub-task of extraction (Downey et al., 2007). The typechecking task is to rank extractions with arguments that are of the proper type for a relation above type errors. As an example, the extraction `Founded(Bill Gates, Oracle)` is type correct, but is not correct for the extraction task.

2.1 Statistical Language Models

A *Statistical Language Model* (SLM) is a probability distribution $P(\mathbf{w})$ over word sequences $\mathbf{w} = (w_1, ..., w_r)$. The most common SLM techniques are *n-gram models*, which are Markov models in which the probability of a given word is dependent on only the previous $n-1$ words. The accuracy of an n-gram model of a corpus depends on two key factors: the choice of n, and the *smoothing* technique employed to assign probabilities to word sequences seen infrequently in training. We experiment with choices of n from 2 to 4, and two popular smoothing approaches, Modified Kneser-Ney (Chen and Goodman, 1996) and Witten-Bell (Bell et al., 1990).

Unsupervised *Hidden Markov Models* (HMMs) are an alternative SLM approach previously shown to offer accuracy and scalability advantages over n-gram models in ADS (Downey et al., 2007). An HMM models a sentence $\mathbf{w}$ as a sequence of observations w_i each generated by a hidden state variable t_i. Here, hidden states take values from $\{1, \ldots, T\}$, and each hidden state variable is itself generated by some number k of previous hidden states. Formally, the joint distribution of a word sequence $\mathbf{w}$ given a corresponding state sequence $\mathbf{t}$ is:

$$P(\mathbf{w}|\mathbf{t}) = \prod_i P(w_i|t_i)P(t_i|t_{i-1}, \ldots, t_{i-k}) \quad (1)$$

The distributions on the right side of Equation 1 are learned from the corpus in an unsupervised manner using Expectation-Maximization, such that words distributed similarly in the corpus tend to be generated by similar hidden states (Rabiner, 1989).

2.2 Performing ADS with SLMs

The *Assessment by Distributional Similarity* (ADS) technique is to rank extractions in U_R in decreasing order of distributional similarity to the seeds, as estimated from the corpus. In our experiments, we utilize an ADS approach previously proposed for HMMs (Downey et al., 2007) and adapt it to also apply to n-gram models, as detailed below.

Define a *context* of an extraction argument e_i to be a string containing the m words preceding and m words following an occurrence of e_i in the corpus. Let $C_i = \{c_1, c_2, ..., c_{|C_i|}\}$ be the union of all contexts of extraction arguments e_i and seed arguments s_i for a given relation R. We create a *probabilistic context vector* for each extraction e_i where the j-th dimension of the vector is the probability of the context surrounding given the extraction, $P(c_j|e_i)$, computed from the language model. [1]

We rank the extractions in U_R according to how similar their arguments' contextual distributions, $P(c|e_i)$, are to those of the seed arguments. Specifically, extractions are ranked according to:

$$f(e) = \sum_{e_i \in e} KL(\frac{\sum_{w' \in S_{Ri}} P(c|w')}{|S_{Ri}|}, P(c|e_i)) \quad (2)$$

where KL represents KL Divergence, and the outer sum is taken over arguments e_i of the extraction e.

For HMMs, we alternatively rank extractions using the HMM *state distributions* $P(t|e_i)$ in place of the probabilistic context vectors $P(c|e_i)$. Our experiments show that state distributions are much more accurate for ADS than are HMM context vectors.

3 Experiments

In this section, we present experiments showing that SLM accuracy correlates strongly with ADS performance. We also show that SLM performance can be used for model selection, leading to an ADS technique that outperforms previous results.

3.1 Experimental Methodology

We experiment with a wide range of n-gram and HMM models. The n-gram models are trained using the SRILM toolkit (Stolcke, 2002). Evaluating a

[1] For example, for context c_j = "I visited ___ in July" and extraction e_i = "Boston," $P(c_j|e_i)$ is P("I visited Boston in July") / P("Boston"), where each string probability is computed using the language model.

LM	Unary	Binary	Wikipedia
HMM 1-5	-.911	-.361	-.994
HMM 2-5	-.856	.120	-.930
HMM 3-5	-.823	-.683	.922
HMM 1-10	-.916	-.967	-.905
HMM 2-10	-.877	-.797	-.963
HMM 3-10	-.957	-.669	-.924
HMM 1-25	-.933	-.850	-.959
HMM 1-50	-.942	-.942	-.947
HMM 1-100	-.896	-.877	-.942
N-Gram	-.512	-.999	.024

Table 1: Pearson Correlation value for extraction performance (in AUC) and SLM performance (in perplexity). Extraction accuracy increases as perplexity decreases, with an average correlation coefficient of -0.742. "HMM k-T" denotes an HMM model of order k, with T states.

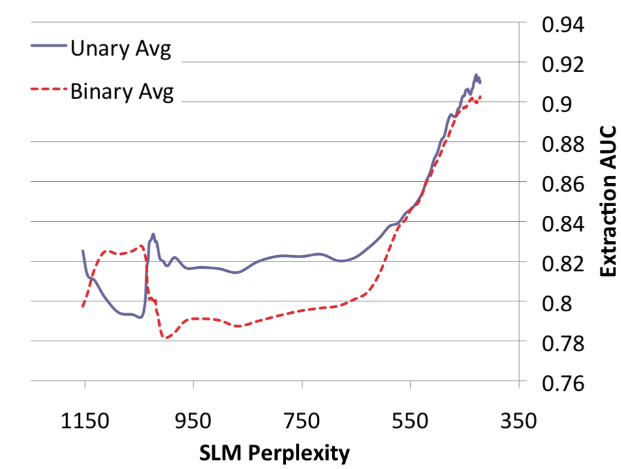

Figure 1: HMM 1-100 Performance. Information Extraction performance (in AUC) increases as SLM accuracy improves (perplexity decreases).

variety of HMM configurations over a large corpus requires a scalable training architecture. We constructed a parallel HMM codebase using the Message Passing Interface (MPI), and trained the models on a supercomputing cluster. All language models were trained on a corpus of 2.8M sentences of Web text (about 60 million tokens). SLM performance is measured using the standard perplexity metric, and assessment accuracy is measured using area under the precision-recall curve (AUC), a standard metric for ranked lists of extractions. We evaluated performance on three distinct data sets. The first two data sets evaluate ADS for unsupervised information extraction, and were taken from (Downey et al., 2007). The first, Unary, was an extraction task for unary relations (Company, Country, Language, Film) and the second, Binary, was a type-checking task for binary relations (Conquered, Founded, Headquartered, Merged). The 10 most frequent extractions served as bootstrapped seeds. The two test sets contained 361 and 265 extractions, respectively. The third data set, Wikipedia, evaluates ADS on weakly-supervised extraction, using seeds and extractions taken from Wikipedia 'List of' pages (Pantel et al., 2009). Seed sets of various sizes (5, 10, 15 and 20) were randomly selected from each list, and we present results averaged over 10 random samplings. Other members of the seed list were added to a test set as correct extractions, and elements from other lists were added as errors. The data set included 2264 extractions across 36 unary relations, including Composers and US Internet Companies.

3.2 Optimizing Language Models for IE

The first question we investigate is whether optimizing individual language models leads to better performance in ADS. We measured the correlation between SLM perplexity and ADS performance as training proceeds in HMMs, and as n and the smoothing technique vary in the n-gram models. Table 1 shows that as the SLM becomes more accurate (i.e. as perplexity decreases), ADS performance increases. The correlation is strong (averaging -0.742) and is consistent across model configurations and data sets. The low positive correlation for the n-gram models on Wikipedia is likely due to a "floor effect"; the models have low performance overall on the difficult Wikipedia data set. The lowest-perplexity n-gram model (Mod Kneser-Ney smoothing with n=3, KN3) does exhibit the best IE performance, at 0.039 (the *average* performance of the HMM models is more than twice this, at 0.084). Figure 1 shows the relationship between SLM and ADS performance in detail for the best-performing HMM configuration.

3.3 Model Selection

Different language models can be configured in different ways: for example, HMMs require choices for the hyperparameters k and T. Here, we show that

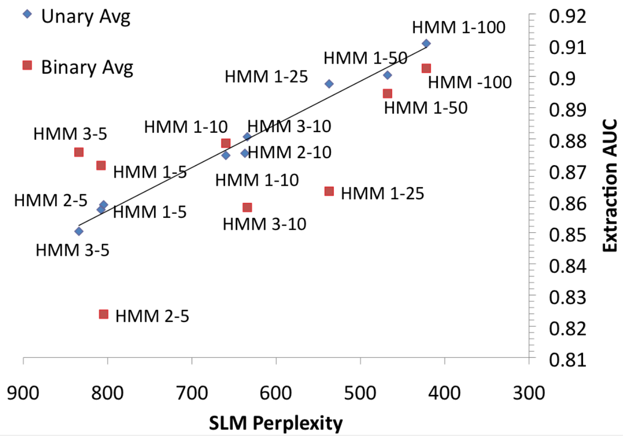

Figure 2: Model Selection for HMMs. SLM performance is a good predictor of extraction performance across model configurations.

Relation	HMM-T	Best HMM
Company	.966	**.985**
Country	.886	**.942**
Languages	**.936**	.914
Film	**.803**	.801
Unary Avg	.898	**.911**
Conquered	.917	**.923**
Founded	**.827**	.799
Merged	.920	**.925**
Headquartered	.734	**.964**
Binary Average	.849	**.903**

Table 2: Extraction Performance Results in AUC for Individual Relations. The lowest-perplexity HMM, 1-100, outperforms the HMM-T model from previous work.

SLM perplexity can be used to select a high-quality model configuration for ADS using only unlabeled data. We evaluate on the Unary and Binary data sets, since they have been employed in previous work on our corpora. Figure 2 shows that for HMMs, ADS performance increases as perplexity decreases across various model configurations (a similar relationship holds for n-gram models). A model selection technique that picks the HMM model with lowest perplexity (HMM 1-100) results in better ADS performance than previous results. As shown in Table 2, HMM 1-100 reduces error over the HMM-T model in (Downey et al., 2007) by 26%, on average. The experiments also reveal an important difference between the HMM and n-gram approaches. While KN3 is more accurate in SLM than our HMM models, it performs *worse* in ADS on average. For example, HMM 1-25 underperforms KN3 in perpexity, at 537.2 versus 227.1, but wins in ADS, 0.880 to 0.853. We hypothesize that this is because the latent state distributions in the HMMs provide a more informative distributional similarity measure. Indeed, when we compute distributional similarity for HMMs using probabilistic context vectors as opposed to state distributions, ADS performance for HMM 1-25 decreases to 5.8% *below* that of KN3.

4 Conclusions

We presented experiments showing that estimating distributional similarity with more accurate statistical language models results in more accurate extraction assessment. We note that significantly larger, more powerful language models are possible beyond those evaluated here, which (based on the trajectory observed in Figure 2) may offer significant improvements in assessment accuracy.

References

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

T. C. Bell, J. G. Cleary, and I. H. Witten. 1990. *Text Compression*. Prentice Hall, January.

Stanley F. Chen and Joshua Goodman. 1996. An empirical study of smoothing techniques for language modeling. In *Proc. of ACL*.

D. Downey, S. Schoenmackers, and O. Etzioni. 2007. Sparse information extraction: Unsupervised language models to the rescue. In *Proc. of ACL*.

Z. Harris. 1985. Distributional structure. In J. J. Katz, editor, *The Philosophy of Linguistics*.

M. Paşca, D. Lin, J. Bigham, A. Lifchits, and A. Jain. 2006. Names and similarities on the web: Fact extraction in the fast lane. In *Procs. of ACL/COLING 2006*.

P. Pantel, E. Crestan, A. Borkovsky, A. M. Popescu, and V. Vyas. 2009. Web-scale distributional similarity and entity set expansion. In *Proc. of EMNLP*.

L. R. Rabiner. 1989. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

R. Snow, D. Jurafsky, and A. Y. Ng. 2006. Semantic taxonomy induction from heterogenous evidence. In *COLING/ACL 2006*.

Andreas Stolcke. 2002. SRILM – an extensible language modeling toolkit. In *Proceedings of ICSLP*, volume 2.