Thesis Proposal

January 12, 2014

I intend to devise a variant of Dan Klein's unsupervised dependency parser "Dependency Model With Valence" a (Klein and Manning, 2004) [1].

1 Parsing

Parsing is one of the fundamental areas in natural language processing. A sentence can have many syntactic trees. The parser builds a model which scores each of these trees and outputs the one with the highest score. Thus, parsing is mapping a sentence to the most appropriate syntactic structure.

1.1 Dependency Parsing

A dependency parse of a sentence is a directed acyclic graph rooted at the special symbol ROOT. The graph should also be projective, implying that no two arcs can cross each other. It is comprised of a series of dependencies. A dependency is an ordered pair of (head, modifier) words. Every word but for the ROOT can have only one head word in all. The ROOT of the whole parse can never be a modifier.

1.2 Why build parsers

Parsing has several applications. It is used to check the grammatical correctness of a sentence. If the sentence complies with the grammar devised by the parser, then it is grammatically correct. It is also used as an intermediate stage in semantic analysis for question answering and relationship extraction. Parsing is the basis for syntactic machine translation.

1.3 Advantages of unsupervised dependency parser

When a parser learns from training data, comprised of the sentences and their associated best dependency parse, it is called supervised parsing. Supervised parsing involves human annotators annotating each sentence with the best dependency parse. It is time consuming and expensive. Human annotators are also prone to errors. There are a lot of languages where the annotated data is unavailable. Unsupervised parsing helps the parser build a model by giving unannotated data. It emphasizes on the redundancy of the patterns in the data. Owing to the aforementioned drawbacks of relying on annotated data, there is active research going on in the field of unsupervised parsing.

2 Dependency Model with Valence

Dependency Model with Valence is an unsupervised model of dependency parsing which has the following steps:

- First, the root of the sentence is generated and it further generates its dependents.
- For each node, all the right dependents are generated to begin with. After all the dependents are generated on the right, a STOP symbol is generated. This STOP symbol indicates that the current word no longer takes any arguments in the present direction. This is followed by the generation of all the left dependents and a STOP symbol on the left.

- Every time before a dependent is generated in a particular direction, a decision is made if the STOP symbol should be generated or it should continue generating dependents. The probability of generating a STOP symbol next is conditioned on the identity of the head and the direction of the attachment and the adjacency (adj) P_{STOP}(¬STOP | h, dir, adj). Adjacency indicates whether or not a dependent is the first modifier of the head word in the current direction.

- A head word takes a dependent in a particular direction conditioned on the head word itself and the direction in which the dependent is taken P_{CHOOSE}(a|h, dir). This entire process is recursive.

For a dependency structure D, let each word h have left dependents $deps_D$ (h, l) and right dependents $deps_D$ (h, r). The probability of the fragment D(h) of the dependency tree rooted at h is given by:

$$P(D(h)) = \prod_{dir=(l,r)} \prod_{a=deps_D(h,dir)} P_{STOP}(\neg STOP|h, dir, adj)$$
$$P_{CHOOSE}(a|h, dir)P(D(a))P_{STOP}(STOP|h, dir, adj)$$

3 Existing variants of DMV

Smith and Eisner (2005) [4] use contrastive estimation together with DMV. Their learner takes into account not only the observed positive examples, but also a set of similar examples that are down-weighted because they could have been observed but were not. Cohen et al. (2008) [3] use Dirichlet priors on the rewriting operations, which can encourage sparse solutions, a property which is important for grammar induction. They derive a variational EM algorithm for the probability estimation and achieve a 59.4% directed attachment score on WSJ10.

Headden et al. (2009) [7] extend the term of valence in DMV and call it Extended Valence Grammar (EVG). The main difference is that generating a new argument is conditioned by the fact whether it is the first one in the given direction or not. The probability P_{CHOOSE} (a|h, dir) in the above equation is thus substituted by P_{CHOOSE} (a|h, dir, adj). Headden et al. also used the lexicalization (the generated arguments are conditioned not only the head part-of-speech but also its word form) and smoothing by interpolation increasing a directed attachment score of 68.9%.

Spitkovsky et al. (2011) [6] observed a strong connection between English punctuation and phrase boundaries, split sentences at punctuation marks and imposed parsing restrictions over their fragments.

4 Initialization of DMV

The Expectation Maximization algorithm maximizes its objective function locally. Probabilistic dependency grammars have several local maxima. One of the most important factors in avoiding EM getting stuck in a local maxima is its initialization. DMV uses an **ad-harmonic initializer**.

Consider a sentence with words $w_1 \ldots w_n$ where n is the number of words in the sentence.

(1) Each word has a uniform probability of becoming a ROOT.

$$P(ROOT) = \frac{1}{n}$$

(2) The probability of dependency between two words is inversely proportional to the distance between them.

$$1 \leq j \leq n\, sum[w_j] = \sum_{1 \leq i \leq n \, and \, i \neq j} \frac{1}{|j-i|}$$

$$j < i \leq n\, P(w_i|w_j, right) = \frac{(n-1)}{n} * \frac{1}{sum[w_j]} * \frac{1}{|j-i|}$$

$$i < j \leq n\, P(w_i|w_j, left) = \frac{(n-1)}{n} * \frac{1}{sum[w_j]} * \frac{1}{|j-i|}$$

The initializer is built under the linguistic intuition that shorter dependencies are preferable to longer. Many techniques have been conceived to help EM achieve better likelihood, two of which that are relevant to us are discuss in this section.

Smith(2006) [5] proposed "The Skewed Deterministic Annealing and Structural Annealing techniques" where the initial parameter settings are biased to reflect this intuition that short dependencies are better. This bias is slowly removed over the course of learning. Deterministic Annealing leads to flatter likelihood surface. This eventually helps in finding maxima with higher likelihood.

Headen et al (2009) [7] used 100 instances of estimating DMV using using Variational Bayes, where each instance is given 20 random restarts. Each restart was run for 40 iterations, and the model with the highest lower bound value was run until convergence. His results indicate that directed accuracy of DMV using Variational Bayes with random initialization has increased by 6% and undirected accuracy increased by 2%.

5 Extending DMV

In order to overcome the problem of local maxima, we intend to use several hundred random restarts for DMV. The parameters for the random restart are generated by drawing uniformly at random from the interval [0,1] and then normalizing. We would like to observe the relationship between the likelihood of the objective function and the accuracy of the parser in each case. Even after several hundred restarts, em will search for model with a better likelihood. The best model generated by random restarts is then compared with the one produced by initializing em with the harmonic initializer.

One of the key challenges faced here is that, as observed by Liang and Klein (2008) [2], as the number of iterations of EM increase, though the likelihood increases, the accuracy decreases. Another one is the enormous increase in the computational power and time needed to run these random initializations.

References

[1] Dan Klein and Christopher Manning. Corpus-based induction of syntactic structure: models of dependency and constituency. In *ACL*, 2004.

[2] Percy Liang and Dan Klein. Analyzing the errors of unsupervised learning. In *ACL*, 2008.

[3] Shay B. Cohen, Kevin Gimpel and Noah A. Smith. Logistic normal priors for unsupervised probabilistic grammar induction. *In Neural Information Processing Systems*, pages 321–âĂŞ328, 2008.

[4] Noah Smith and Jason Eisner. Corpus-based induction of syntactic structure: models of dependency and constituency. In *JCAI Workshop on Grammatical Inference Applications*, pages 73–82, 2005.

[5] Noah Smith and Jason Eisner. Annealing structural bias in multilingual weighted grammar induction. In *ACL*, pages 569–576, 2006.

[6] Valentin I. Spitkovsky, Hiyan Alshawi and Daniel Jurafsky. Punctuation: Making a point in unsupervised dependency parsing. 2011.

[7] William P. Headden, III, Mark Johnson, and David McClosky. Improving unsupervised dependency parsing with richer contexts and smoothing. In *ACL*, pages 101–109, 2009.