Refining NELL's Internet-Extracted Knowledge Base in a Scalable, Distributed Fashion (Midterm Report)

Paul Elliott
University of Oregon
paule@cs.uoregon.edu

David Stevens
University of Oregon
dstevens@cs.uoregon.edu

Abstract

Our project involves refining noisy structured knowledge bases in a scalable, distributed manner. We focus on NELL, the Never Ending Language Learner, which extracts information from the Web and compiles it into a belief ontology. We will leverage GraphLab's, parallel belief propagation algorithm to correct the uncertainty in NELL's knowledge base. In doing so, we hope to provide a distributed solution to such noise-correction tasks that performs more scalably and efficiently than current solutions.

1 Activities

In this section, we cover the work we have done in the first half of the term. At this point in the project, we are on schedule with respect to our timeline. We began with focused literature review. We first reviewed machine learning inference algorithms including Markov logic networks and belief propagation, in order to understand the general procedure for cleaning up a noisy knowledge base. We used Shangpu's paper submission on refining NELL's [1] knowledge base with the Markov Chain Monte Carlo algorithm as reference, as well as web resources and conversations with Shangpu and Professor Daniel Lowd.

In addition to literature review, we also began familiarizing ourselves with GraphLab [2]. We downloaded the GraphLab repository to our local machines, and compiled and built the framework. We then investigated GraphLab's graphical models toolkit, which provides the parallel loopy belief propagation algorithm for refining structured noisy data such as NELL's knowledge base. We examined the required source file formats and analyzed the toolkit's source code for the inference algorithm, and finally ran a successful test on a sample dataset.

After working through the literature review and the GraphLab framework, we began our design and implementation phase on schedule in week five. We obtained our dataset of NELL's extracted knowledge base, which was then preprocessed to ground all first-order logic clauses into independent variables. Next, we built a Python script to parse the corrected dataset and convert it into the vertex and edge datasets (beliefs and relations between beliefs, respectively) necessary for the graphical models toolkit. Finally, we ran a successful initial test on a single machine, and observed the results.

In the following section, we detail the issues and findings we encountered during the first half of our work.

2 Findings

During the first five weeks of the term, we uncovered some interesting issues with our project. First, we found some inconsistencies within our dataset. In addition to trivial errors such as edge weights without edge definitions–which we ignore– we found and addressed the following issues:

1. *Duplicate vertices with different confidence values.* This is likely due to degrading confidence over time, and therefore we use the lowest (most recent) confidence value and will investigate next week.
2. *Hard constraints, representing manually verified facts.* We give these vertices a confidence value of 1, and note that we will have to address hard constraints in our inference algorithm.
3. *Conflicting beliefs x and $\neg x$.* The vertex datasets contain a probability distribution of all possible values (in our case, true or false). We simply combine the conflicting beliefs into a single entry, and give precedence to hard constraints as above.
4. *Existing edge data containing non-existent vertices.* This is caused by manually verified facts about vertices that do not exist in the knowledge base. We subsequently ignore them.

Our results from the initial test look promising. Our corrected knowledge base shows considerable updates of confidence values, and we observe increases in confidence values on verifiably correct beliefs. We have also observed a few decreases in confidence values that are questionable, and will look into these cases in the beginning of next week.

3 Future Work

Having discussed what we have done in the first half of this project, we now address our plan for the remainder of the term. Our updated timeline is as follows:

- **Week 6** Address remaining issues with dataset inconsistencies, and begin porting our solution to run on the ACIS grid.
- **Week 7** Address any arising issues with parallelizing our solution, including how to properly partition the inference task.
- **Week 8** Define metrics for experiments, run experiments, collect data.
- **Week 9** Finish remaining experiment and data collection tasks, begin analyzing results.
- **Week 10** Formalize results and analysis, write paper, present research.

4 Discussion

In our work thus far, we have arrived at some helpful takeaways for future work. For one, we note that many issues have arisen from inconsistencies in NELL's knowledge base. It is clear that in conducting research, one must not assume perfect correctness in the data sources used, and one must be able to handle to inconsistencies in such sources as they arise. Following this caveat, we also find that it is important to thoroughly examine your data sources as we have done.

In evaluating our work at its halfway point, we conservatively give ourselves a grade of A-. We have completed milestones in accordance with our timeline, and have successfully accomplished our base step of running the inference algorithm on our corrected dataset on a single machine. With this work behind us, we are well staged to begin parallelizing our solution and testing it on the ACIS grid. Moreover, we have learned a considerable amount of information about GraphLab, one of the more capable distributed machine learning frameworks available today. In leveraging this work toward extracting structured knowledge from the Internet, we contribute to a valuable task involving the most influential distributed system of our time.

References

[1] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E. Hruschka Jr, and T. Mitchell. Toward an architecture for never-ending language learning. In *Proceedings of the Twenty-Fourth Conference on Artificial Intelligence (AAAI 2010)*, volume 2, pages 3–3, 2010.

[2] Y. Low, J. Gonzalez, A. Kyrola, D. Bickson, C. Guestrin, and J. Hellerstein. Graphlab: A new framework for parallel machine learning. *arXiv preprint arXiv:1006.4990*, 2010.