Illinois Named Entity Recognizer: Addendum to Ratinov and Roth '09 reporting improved results

Tom Redman, Mark Sammons, and Dan Roth
Cognitive Computation Group, University of Illinois at Urbana-Champaign
{t-redman,mssammon,danr}@illinois.edu

1 Updates to Named Entity Recognizer software

Since the release of the Illinois Named Entity Recognizer described in [2], we have made a number of improvements to the software. Modifications were made to improve reliability, reduce memory footprint and improve wall clock time performance. Several bugs were fixed long the way, and the gazetteers were updated, improving upon what was already state-of-the-art accuracy.

1.1 Memory footprint

The memory footprint is significantly smaller as the result of stripping the core token representation to a much leaner representation. Unused fields were removed from this data structure. Additionally, the code base was surveyed to ensure static data was used appropriately and in a read-only fashion, and that such data was never replicated. Many large data structures were also optimized resulting in both performance and memory footprint improvements.

Performance was also improved by a new gazetteer look-up implementation. Although this slightly increased the memory footprint, the new implementation improved flexibility (this new gazetteer can find phrases of any length, where the previous implementation limited the size of phrases) and significantly improved performance.

1.2 Scalability

There was also a keen focus on scalability issues in this version. Previous version were thread-safeat the expense of multiprocessor performance. This current implementation has run on many cores simultaneously with considerably less resource contention. This work has produced a version limited more by the bandwidth of the system bus than by access to shared resources.

1.3 API

This version is using the standard View/Constituent API from the Illinois Cognitive Computation Group's core libraries (https://github.com/IllinoisCogComp/illinois-cogcomp-nlp). This simple API allows users to make a call resulting in a data structure that is easy to parse and understand. We have also included an improved interactive menu driven command line utility that can be used to process files in bulk or individually. The configuration file has been dramatically simplified further improving the user experience.

2 Evaluation

Table 2 collects statistics on the performance of the updated NER package and compares it to the figures reported in [2]. The evaluation on OntoNotes [1] has been added for completeness. Memory overhead was not recorded for the original software release.

Version	Ratinov & Roth 2009	2016 Release
CoNLL F1	90.57	91.06
MUC F1	85.62	88.31
Web F1	74.53	79.50
OntoNotes F1	-	84.63
Memory Overhead	-	1.2G

Table 1: **Evaluation of the new Illinois NER software, compared to its previous performance.** Note that the different data sets listed have different label sets and annotations, so numbers are not comparable across tasks. For details, see [2].

References

[1] E. Hovy, M. Marcus, M. Palmer, L. Ramshaw, and R. Weischedel. Ontonotes: The 90% solution. In *Human Language Technologies - North American Chapter of the Association for Computational Linguistics*, New York, 2006.

[2] L. Ratinov and D. Roth. Design challenges and misconceptions in named entity recognition. In *Proc. of the Conference on Computational Natural Language Learning (CoNLL)*, 6 2009.