1 Introduction

Although clusters of commodity hardware are commonplace in companies, research labs, and are even available to the public through services such as Amazon.com’s EC2 for as little as pennies per hour, the open source model-estimation tools available to the statistical machine translation community, such as the Moses toolkit,\(^1\) are designed for single-core execution. Unfortunately, this is hardly optimal: generating a word alignment for a million-sentence parallel corpus using GIZA++ takes 24 hours on a single core, and building a phrase-based translation model from 5M sentences (the amount of Arabic-English training data available from the LDC) with the Moses tools takes just shy of 3 days [2]. These latencies are a source of frustration to many researchers. Worse still, they have had a chilling effect on the kind of research that is being carried out by the SMT community: experimental manipulations that occur “up-stream” from the word-alignment or phrase-extraction processes tend to be evaluated on artificially small amounts of training data (e.g., [6, 7, 9]). Finally, as data sizes continue to grow, some single-core tools, such as GIZA++ do not even appear able to scale, and researchers must resort to ad-hoc solutions, such as dividing the training data. In recent research, we have demonstrated that the MapReduce paradigm [1], offers a compelling solution to the problem of how to let the algorithms used to compute word alignment and to estimate translation models take advantage of the computational power of a full cluster of commodity hardware [2]. On sufficiently large corpora, we can see nearly optimal speed-ups in training time as nodes are added to the cluster for both EM training and phrase-model estimation, which we believe will be of tremendous value to the research community. We therefore request funding to enhance our research-quality software and release it as a fully open-source model-estimation suite that is built on top of Hadoop,\(^2\) an open-source implementation of Google’s GFS and MapReduce frameworks [3, 1]. This will enable researchers to make better use of existing computational resources as the amount of training data available continues to grow.

2 Detailed Proposal

We have currently developed:

- A MapReduce tool that implements Model 1, Model 2, and HMM word alignment using fully distributed EM training as described in [2].

- A MapReduce tool that can perform standard approaches to alignment symmetrization [5].

---

\(^1\)http://www.statmt.org/moses

\(^2\)http://hadoop.apache.org/
• A MapReduce tool that estimates phrase-based translation models, equivalent to the current Moses implementation, as described in [2].

If funded, we plan to provide:

• A MapReduce tool to efficiently compile a bitext into the binary representation used by our tools.
• An augmented word aligner that implements the initialization heuristics described in [7] and the NULL-word HMM extensions described in [8].
• A MapReduce tool that estimates the lexicalized reordering models currently implemented in the Moses training suite, as described in [4].
• A suite that unifies bidirectional word alignment, phrase extraction and scoring, and MapReduce tool for filtering a model (stored in Hadoop’s distributed filesystem) for a particular test set.
• Documentation for using the toolkit with a virtual cluster running on Amazon.com’s EC2.
• Documentation for using the toolkit with Hadoop-on-Demand, a tool that lets Hadoop run on top of a cluster running Sun GridEngine, or Torque.
• A website containing documentation, a user mailing list, and public source code repository.

References


