High–Performance Training of Conditional Random Fields for Large–Scale Sequential Labeling Applications

H. Phan[1], **M. Nguyen**[2], **Y. Inoguchi**[3], **B. Ho**[4], **and S. Horiguchi**[5]

Abstract: Labeling and segmenting for complex data have become significant research issues in machine learning, natural language processing, and computational biology. The problems of identifying part–of–speech, phrase types in natural language, and predicting secondary structures of protein sequences are fundamental research tasks in their fields. Conditional Random Fields (CRFs), first introduced by Lafferty et al. (2001), are deliberately designed to deal with such problems. CRFs have the ability to encode the dependencies among data elements as well as the flexibility to incorporate various features from empirical data. Further, the training of CRFs, performed based on the constrained convex optimization of the log–likelihood function, allows us to obtain the global optimal configuration.

However, the current training methods for CRFs are time–consuming due to the repeatedly expensive evaluation of likelihood function and its gradient vector. This is because real–world applications have to deal with very high–dimensional data, and thus the objective function (i.e., the log–likelihood function) to be optimized might have hundreds of thousands or millions parameters. It usually takes several days to train CRFs on large–scale datasets. Fortunately, the nature sum of the log–likelihood function allows us to think of a potential parallel training strategy for CRFs. In this paper, we present a high–performance parallel implementation of CRFs on massively parallel systems. In our parallel algorithm, the training dataset is randomly divided into equal partitions; the log–likelihood function and its gradient vector are evaluated in parallel; the root process then gathers those values in order to compute the new setting of parameters (this computation is based on a quasi–Newton method, L–BFGS [Liu & Nocedal 1989]). Finally, the root process broadcasts the new parameter setting to all the others for the next training iteration. The algorithm stops when it reaches the global optimum or after a finite number of iterations (specified by users). The experimental results for phrase segmentation on PennTree Bank (a very large linguistic dataset) using a Cray XT3 system (90PEs, 4 x 2.4GHz CPUs and 32GB RAM on each PE) and Message Passing Interface (MPI), showed that our parallel implementation can reduce the computational time dramatically. Furthermore, the powerful computing resource allows us to build and test the second–order Markov CRFs and achieved significantly higher accuracy in comparison with the state–of–the–art results.

[1,2,3,4] Japan Advanced Institute of Science and Technology
1-1, Asahidai, Nomi, Ishikawa, 923–1292 Japan
{*hieuxuan, nguyenml, inoguchi, bao*}*@jaist.ac.jp*

[5] Tohoku University
Aoba 6–3–09 Sendai, 980–8579 Japan
susumu@ecei.tohoku.ac.jp