Better Chinese Word Segmentation via Dual Decomposition (Title to be changed)

Author 1
XYZ Company
111 Anywhere Street
Mytown, NY 10000, USA
author1@xyz.org

Author 2
ABC University
900 Main Street
Ourcity, PQ, Canada A1A 1T2
author2@abc.ca

Abstract

1 Introduction

First paragraph talks about why Chinese Word Segmentation is important.

Second paragraph talks about the challenges, specifically ambiguity and unknown words.

Third paragraph: There are two classes of models: character-based (Xue, 2003; Tseng et al., 2005; Zhang et al., 2006; Wang et al., 2010) and word-based (Andrew, 2006; Zhang and Clark, 2007), each has their respective advantage and disadvantage: char-based segmenters models the internal structure of word composition better and therefore are more effective at detecting out-of-vocabulary words, however they suffer from inconsistency issues. Word-segmenters are better at memorizing the training word type lexicon, and have better consistency, but not as good at detecting OOV words. An additional advantage of word-segmenter is that they can explore larger context than character-based models (this is a little hard to explain without getting into the details of how these models work, maybe leave this piece till later; the basic idea is that character-based models – i.e., linear-chain CRFs — have very limited context due to Markov assumption, it typically only have direct information about the immediate neighboring characters, although information from longer-range context does flow because of the sentence-level normalization, its effect is much less direct. Word-based models in contrary directly captures the formation of entire words, say the previous word token has 3 characters, and the current word being considered has 3 characters, then a word-level bigram feature would capture context in neighboring 6 characters.

Fourth paragraph (maybe consider moving some of this into a related work section and put it at the end): various mixing approaches have been proposed to combine the merits of these two classes of models (Wang et al., 2006; Lin, 2009; Sun et al., 2009; Sun, 2010; Wang et al., 2010).

(Sun et al., 2009) is the most similar in that they also try to directly combine two segmenters – one char-based and one word-based. But they do it through segmenter bagging, which would require training up 50 or more individual segmenters and poll their results at test time, significantly increases training and testing effort, not practical. (Sun et al., 2009) presented a latent-variable model. very complicated, hard to train. our method is much simpler and works better than their method.

(Lin, 2009) and (Wang et al., 2006) both try to incorporate a character-based local classifier model (MaxEnt) with a language model that captures more word-level context. In order to bring in bigram language model, (2009) gave a heuristic decoding method that involves various conditioning and back-off; (2006) gave a modified Viterbi algorithm actually has complexity of $O(T^3)$.

In this work, we propose a simple and principled joint decoding method for combining character-based and word-based segmenter based on dual decomposition. This method has strong optimality guarantees and works very well empirically. It's easy to implement and does not require retraining of existing character and word segmenters. Experimental results on standard SIGHAN 2003 and 2005 bake-off evaluations show that our model consistently outperform the word and character baselines by significant margin on all 8 datasets. In particular, it improves OOV recall rates and consistency (yet to be validated pending consistency experiment results). It gives the best reported results to date on 6 out of 8 datasets.

2 Char-based vs. Word-based Segmenters

2.1 Char-based models

Describe models that treat CWS as a character sequence tagging task (first proposed by (Xue, 2003)), widely adopted and most commonly used these days. Give the equation of CRF (see (Tseng et al., 2005)). Briefly describe the features and point out how some of them can be expected to generalize well to OOV words (e.g. suffix and prefix features).

2.2 Word-based models

See (Sun, 2010) for a good description. We used the perceptron based model proposed by (Zhang and Clark, 2007), give the equation of the perceptron, and describe the features. Point out how some of these features basically directly capture the training lexicon.

2.3 Relative Strength and Weakness

Char-based CRF models are better at generalizing over to OOV words (thus higher OOV recall). (NOTE: 6 out of 8 datasets on test, CRF has better OOV recall, but we should defer to talk about this in the results section). But as a tradeoff, it generates more inconsistent segmentation, and creates lots of spurious words. The advantage of Word-based segmenters are that they are more consistent with the training lexicon, words seen in training lexicon are not likely to be segmented into many new ways; disadvantage is that it doesn't generalize well to OOV. Also exact inference is difficult, need to resolve to approximate decoding algorithms such as beam-searching.

3 Dual-Decomposition

Dual decomposition offers a ideal framework for combining these two sources of signals without incurring high cost in model complexity (in contrast to (Sun et al., 2009)) or decoding efficiency (in contrast to bagging in (Wang et al., 2006; Sun, 2010)). DD has been successfully applied to similar situations where we want to combine local model with global models, for example, in dependency parsing (Koo et al., 2010)), bilingual sequence tagging (Wang et al., 2013) and word alignment (). Give a brief description of DD algorithm, focus on the intuition. See (Wang et al., 2013) and (DeNero and Macherey, 2011) for a good short introduction example. Refer users to (Rush and Collins, 2012) for a full tutorial on dual decomp. The modification to Viterbi decoding is exactly the same as in (Wang et al., 2013) and (DeNero and Macherey, 2011). The modification to the beam-search is similar, each time we extend a hypothesis with a new character, depending if the new character is appended to the last word or starting a new word, the corresponding DD penalty is factored into the score for the new hypothesis.

4 Experiments

Give URL for the Stanford CRF segmenter. The perceptron-based word segmenter is a re-implementation of (Zhang and Clark, 2007). For development data, we used: Training: CTB section 1-270 400-931 1001-115; Dev: CTB section 271-300.

Hyper-parameters tuned on the dev set. Perceptron: 10 iterations of training. CRF: L2-regularization Sigma set to 3. Dual-decomp: initial step size set to 0.1, 100 iterations.

4.1 Dataset

Give description of the SIGHAN 2003 (Sproat and Emerson, 2003) and 2005 (Emerson, 2005) bake-off datasets. List a table of the corpora names, size of training, testing, etc, similar to (Sun, 2010).

5 Results

Reformat the table so it fits in a single column. The OOV rates should be moved out of this table and put in the data description tables in Experiment section Dataset subsection.

6 Discussion

In this section we talked about detailed consistency analysis and maybe give some examples.

SIGHAN 2005

		Recall	Precision	F_1	OOV	R_{oov}	R_{iv}
AS	SIGHAN winner	0.952	**0.951**	0.952	0.043	0.696	0.963
	CRF-Char	0.952	0.936	0.944	0.043	0.589	0.969
	Perceptron-Word	0.958	0.950	0.954	0.043	0.695	0.970
	Dual Decomp	0.959	0.949	**0.954**	0.043	0.677	**0.972**
PKU	SIGHAN winner	**0.953**	0.946	0.950	0.058	0.636	**0.972**
	CRF-Char	0.946	0.953	0.949	0.058	0.778	0.956
	Perceptron-Word	0.941	0.955	0.948	0.058	0.767	0.952
	Dual Decomp	0.948	**0.957**	**0.953**	0.058	**0.787**	0.958
CITYU	SIGHAN winner	0.941	**0.946**	0.943	0.074	0.698	0.961
	CRF-Char	0.947	0.940	0.943	0.074	**0.761**	0.962
	Perceptron-Word	0.943	0.940	0.942	0.074	0.717	0.961
	Dual Decomp	**0.950**	0.944	**0.947**	0.074	0.753	**0.965**
MSR	SIGHAN winner	0.962	0.966	0.964	0.026	0.717	0.968
	CRF-Char	0.964	0.966	0.965	0.026	0.713	0.971
	Perceptron-Word	0.970	0.972	0.971	0.026	0.746	0.976
	Dual Decomp	**0.973**	**0.974**	**0.974**	0.026	**0.760**	**0.979**
SIGHAN 2003							
AS	SIGHAN winner	0.966	0.956	0.961	0.022	0.364	**0.980**
	CRF-Char	0.969	0.969	0.969	0.022	0.748	0.974
	Perceptron-Word	0.967	0.967	0.967	0.022	0.729	0.972
	Dual Decomp	**0.970**	**0.971**	**0.971**	0.022	**0.775**	0.975
PKU	SIGHAN winner	**0.962**	0.940	0.951	0.069	0.724	**0.979**
	CRF-Char	0.954	0.952	0.953	0.069	0.803	0.965
	Perceptron-Word	0.949	0.952	0.950	0.069	0.790	0.960
	Dual Decomp	0.954	**0.954**	**0.954**	0.069	**0.806**	0.965
CITYU	SIGHAN winner	0.947	0.934	0.940	0.071	0.625	**0.972**
	CRF-Char	0.944	0.939	0.941	0.071	0.741	0.959
	Perceptron-Word	0.944	0.945	0.945	0.071	0.730	0.960
	Dual Decomp	**0.950**	**0.949**	**0.949**	0.071	**0.754**	0.965
CTB	SIGHAN winner	**0.886**	0.875	**0.881**	0.181	0.644	**0.927**
	CRF-Char	0.869	0.865	0.867	0.181	0.680	0.910
	Perceptron-Word	0.865	0.871	0.868	0.181	0.660	0.910
	Dual Decomp	0.876	**0.878**	0.877	0.181	**0.692**	0.917

Table 1: Results on SIGHAN 2005 and 2003 datasets.

6.1 Dual decomposition convergence

Plot the histogram of the number of iterations it takes to converge, and percentage of optimality.

6.2 consistency

give results of the consistency analysis

6.3 oracle

Give results of the oracle experiment

6.4 Error analysis

Maybe look at cases where the model choose the worse of the two instead of the better of the two. See if there are patterns or insights we can draw. Not very important for a short paper.

7 Related Work

If the Introduction section becomes too long, or we have some space left at the end, we can consider move some of the comparison stuff into this section.

8 Conclusion

In this paper we presented ... blah blah. Stress the simplicity and effectiveness of our method.

References

Galen Andrew. 2006. A hybrid Markov/semi-Markov conditional random field for sequence segmentation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

John DeNero and Klaus Macherey. 2011. Model-based aligner combination using dual decomposition. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Thomas Emerson. 2005. The second international Chinese word segmentation bakeoff. In *Proceedings of the fourth SIGHAN workshop on Chinese language Processing*.

Terry Koo, Alexander M. Rush, Michael Collins, Tommi Jaakkola, and David Sontag. 2010. Dual decomposition for parsing with non-projective head automata. In *Proceedings of EMNLP*.

Dekang Lin. 2009. Combining language modeling and discriminative classification for word segmentation. In *Proceedings of the 10th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing)*.

Alexander M. Rush and Michael Collins. 2012. A tutorial on dual decomposition and Lagrangian relaxation for inference in natural language processing. *JAIR*, 45:305–362.

Richard Sproat and Thomas Emerson. 2003. The first international Chinese word segmentation bakeoff. In *Proceedings of the second SIGHAN workshop on Chinese language Processing*.

Xu Sun, Yaozhong Zhang, Takuya Matsuzaki, Yoshimasa Tsuruoka, and Jun'ichi Tsujii. 2009. A discriminative latent variable chinese segmenter with hybrid word/character information. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*.

Weiwei Sun. 2010. Word-based and character-basedword segmentation models: Comparison and combination. In *Proceedings of The 23rd International Conference on Computational Linguistics (COLING)*.

Huihsin Tseng, Pichuan Chang, Galen Andrew, Daniel Jurasfky, and Christopher Manning. 2005. A conditional random field word segmenter for sighan bakeoff 2005. In *Proceedings of the fourth SIGHAN workshop on Chinese language Processing*.

Xinhao Wang, Xiaojun Lin, Dianhai Yu, Hao Tian, and Xihong Wu. 2006. Chinese word segmentation with maximum entropy and n-gram language model. In *Proceedings of the fifth SIGHAN workshop on Chinese language Processing*.

Kun Wang, Chengqing Zong, and Keh-Yih Su. 2010. A character-based joint model for chinese word segmentation. In *Proceedings of The 23rd International Conference on Computational Linguistics (COLING)*.

Mengqiu Wang, Wanxiang Che, and Christopher D. Manning. 2013. Joint word alignment and bilingual named entity recognition using dual decomposition. In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Nianwen Xue. 2003. Chinese word segmentation as character tagging. *International Journal of Computational Linguistics and Chinese Language Processing*, pages 29–48.

Yue Zhang and Stephen Clark. 2007. Chinese segmentation with a word-based perceptron algorithm. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Ruiqiang Zhang, Genichiro Kikui, and Eiichiro Sumita. 2006. Subword-based tagging by conditional random fields for Chinese word segmentation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL (HLT-NAACL)*.