Reducing Approximation and Estimation Errors for Chinese Lexical Processing with Heterogeneous Annotations

Weiwei Sun[†] and **Xiaojun Wan**[‡] [*]
[†‡]Institute of Computer Science and Technology, Peking University
[†]Saarbrücken Graduate School of Computer Science
[†]Department of Computational Linguistics, Saarland University
[†]Language Technology Lab, DFKI GmbH
{ws,wanxiaojun}@pku.edu.cn

Abstract

We address the issue of consuming heterogeneous annotation data for Chinese word segmentation and part-of-speech tagging. We empirically analyze the diversity between two representative corpora, i.e. Penn Chinese Treebank (CTB) and PKU's People's Daily (PPD), on manually mapped data, and show that their linguistic annotations are systematically different and highly compatible. The analysis is further exploited to improve processing accuracy by (1) integrating systems that are respectively trained on heterogeneous annotations to reduce the approximation error, and (2) re-training models with high quality automatically converted data to reduce the estimation error. Evaluation on the CTB and PPD data shows that our novel model achieves a relative error reduction of 11% over the best reported result in the literature.

1 Introduction

A majority of data-driven NLP systems rely on large-scale, manually annotated corpora that are important to train statistical models but very expensive to build. Nowadays, for many tasks, multiple heterogeneous annotated corpora have been built and publicly available. For example, the Penn Treebank is popular to train PCFG-based parsers, while the Redwoods Treebank is well known for HPSG research; the Propbank is favored to build general semantic role labeling systems, while the FrameNet is attractive for predicate-specific labeling. The annotation schemes in different projects are usually different, since the underlying linguistic theories vary and have different ways to explain the same language phenomena. Though statistical NLP systems usually are not bound to specific annotation standards, almost all of them assume homogeneous annotation in the training corpus. The co-existence of heterogeneous annotation data therefore presents a new challenge to the consumers of such resources.

There are two essential characteristics of heterogeneous annotations that can be utilized to reduce two main types of errors in statistical NLP, i.e. the approximation error that is due to the intrinsic suboptimality of a model and the estimation error that is due to having only finite training data. First, heterogeneous annotations are (similar but) *different* as a result of different annotation schemata. Systems respectively trained on heterogeneous annotation data can produce different but relevant linguistic analysis. This suggests that complementary features from heterogeneous analysis can be derived for disambiguation, and therefore the approximation error can be reduced. Second, heterogeneous annotations are (different but) *similar* because their linguistic analysis is highly correlated. This implies that appropriate conversions between heterogeneous corpora could be reasonably accurate, and therefore the estimation error can be reduced by reason of the increase of reliable training data.

This paper explores heterogeneous annotations to reduce both approximation and estimation errors for Chinese word segmentation and part-of-speech (POS) tagging, which are fundamental steps for more advanced Chinese language processing tasks. We empirically analyze the diversity between two representative popular heterogeneous corpora, i.e.

[*]This work is mainly finished when the first author was in Saarland University and DFKI. Both authors are the corresponding authors.

Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics, pages 232–241,
Jeju, Republic of Korea, 8-14 July 2012. ©2012 Association for Computational Linguistics

Penn Chinese Treebank (CTB) and PKU's People's Daily (PPD). To that end, we manually label 200 sentences from CTB with PPD-style annotations.[1] Our analysis confirms the aforementioned two properties of heterogeneous annotations. Inspired by the sub-word tagging method introduced in (Sun, 2011), we propose a structure-based stacking model to fully utilize heterogeneous word structures to reduce the approximation error. In particular, joint word segmentation and POS tagging is addressed as a two step process. First, character-based taggers are respectively trained on heterogeneous annotations to produce multiple analysis. The outputs of these taggers are then merged into sub-word sequences, which are further re-segmented and tagged by a sub-word tagger. The sub-word tagger is designed to refine the tagging result with the help of heterogeneous annotations. To reduce the estimation error, we employ a learning-based approach to convert complementary heterogeneous data to increase labeled training data for the target task. Both the character-based tagger and the sub-word tagger can be refined by re-training with automatically converted data.

We conduct experiments on the CTB and PPD data, and compare our system with state-of-the-art systems. Our structure-based stacking model achieves an f-score of 94.36, which is superior to a feature-based stacking model introduced in (Jiang et al., 2009). The converted data can also enhance the baseline model. A simple character-based model can be improved from 93.41 to 94.11. Since the two treatments are concerned with reducing different types of errors and thus not fully overlapping, the combination of them gives a further improvement. Our final system achieves an f-score of 94.68, which yields a relative error reduction of 11% over the best published result (94.02).

2 Joint Chinese Word Segmentation and POS Tagging

Different from English and other Western languages, Chinese is written without explicit word delimiters such as space characters. To find and classify the basic language units, i.e. words, word segmentation and POS tagging are important initial steps for Chinese language processing. Supervised learning with specifically defined training data has become a dominant paradigm. Joint approaches that resolve the two tasks simultaneously have received much attention in recent research. Previous work has shown that joint solutions led to accuracy improvements over pipelined systems by avoiding segmentation error propagation and exploiting POS information to help segmentation (Ng and Low, 2004; Jiang et al., 2008a; Zhang and Clark, 2008; Sun, 2011).

Two kinds of approaches are popular for joint word segmentation and POS tagging. The first is the "character-based" approach, where basic processing units are characters which compose words (Jiang et al., 2008a). In this kind of approach, the task is formulated as the classification of characters into POS tags with boundary information. For example, the label *B-NN* indicates that a character is located at the begging of a noun. Using this method, POS information is allowed to interact with segmentation. The second kind of solution is the "word-based" method, also known as semi-Markov tagging (Zhang and Clark, 2008; Zhang and Clark, 2010), where the basic predicting units are words themselves. This kind of solver sequentially decides whether the local sequence of characters makes up a word as well as its possible POS tag. Solvers may use previously predicted words and their POS information as clues to process a new word.

In addition, we proposed an effective and efficient stacked sub-word tagging model, which combines strengths of both character-based and word-based approaches (Sun, 2011). First, different character-based and word-based models are trained to produce multiple segmentation and tagging results. Second, the outputs of these coarse-grained models are merged into sub-word sequences, which are further bracketed and labeled with POS tags by a fine-grained sub-word tagger. Their solution can be viewed as utilizing stacked learning to integrate heterogeneous models.

Supervised segmentation and tagging can be improved by exploiting rich linguistic resources. Jiang et al. (2009) presented a preliminary study for annotation ensemble, which motivates our research as well as similar investigations for other NLP tasks,

[1] The first 200 sentences of the development data for experiments are selected. This data set is submitted as a supplemental material for research purposes.

e.g. parsing (Niu et al., 2009; Sun et al., 2010). In their solution, heterogeneous data is used to train an auxiliary segmentation and tagging system to produce informative features for target prediction. Our previous work (Sun and Xu, 2011) and Wang et al. (2011) explored unlabeled data to enhance strong supervised segmenters and taggers. Both of their work fall into the category of feature induction based semi-supervised learning. In brief, their methods harvest useful string knowledge from unlabeled or automatically analyzed data, and apply the knowledge to design new features for discriminative learning.

3 About Heterogeneous Annotations

For Chinese word segmentation and POS tagging, supervised learning has become a dominant paradigm. Much of the progress is due to the development of both corpora and machine learning techniques. Although several institutions to date have released their segmented and POS tagged data, acquiring sufficient quantities of high quality training examples is still a major bottleneck. The annotation schemes of existing lexical resources are different, since the underlying linguistic theories vary. Despite the existence of multiple resources, such data cannot be simply put together for training systems, because almost all of statistical NLP systems assume homogeneous annotation. Therefore, it is not only interesting but also important to study how to fully utilize heterogeneous resources to improve Chinese lexical processing.

There are two main types of errors in statistical NLP: (1) the approximation error that is due to the intrinsic suboptimality of a model and (2) the estimation error that is due to having only finite training data. Take Chinese word segmentation for example. Our previous analysis (Sun, 2010) shows that one main intrinsic disadvantage of character-based model is the difficulty in incorporating the whole word information, while one main disadvantage of word-based model is the weak ability to express word formation. In both models, the significant decrease of the prediction accuracy of out-of-vocabulary (OOV) words indicates the impact of the estimation error. The two essential characteristics about systematic diversity of heterogeneous annotations can be utilized to reduce both approximation and estimation errors.

3.1 Analysis of the CTB and PPD Standards

This paper focuses on two representative popular corpora for Chinese lexical processing: (1) the Penn Chinese Treebank (CTB) and (2) the PKU's People's Daily data (PPD). To analyze the diversity between their annotation standards, we pick up 200 sentences from CTB and manually label them according to the PPD standard. Specially, we employ a PPD-style segmentation and tagging system to automatically label these 200 sentences. A linguistic expert who deeply understands the PPD standard then manually checks the automatic analysis and corrects its errors.

These 200 sentences are segmented as 3886 and 3882 words respectively according to the CTB and PPD standards. The average lengths of word tokens are almost the same. However, the word boundaries or the definitions of words are different. 3561 word tokens are consistently segmented by both standards. In other words, 91.7% CTB word tokens share the same word boundaries with 91.6% PPD word tokens. Among these 3561 words, there are 552 punctuations that are simply consistently segmented. If punctuations are filtered out to avoid overestimation of consistency, 90.4% CTB words have same boundaries with 90.3% PPD words. The boundaries of words that are differently segmented are *compatible*. Among all annotations, only one cross-bracketing occurs. The statistics indicates that the two heterogenous segmented corpora are systematically different, and confirms the aforementioned two properties of heterogeneous annotations.

Table 1 is the mapping between CTB-style tags and PPD-style tags. For the definition and illustration of these tags, please refers to the annotation guidelines[2]. The statistics after colons are how many times this POS tag pair appears among the 3561 words that are consistently segmented. From this table, we can see that (1) there is no one-to-one mapping between their heterogeneous word classification but (2) the mapping between heterogeneous tags is not very uncertain. This simple analysis indicates

[2] Available at http://www.cis.upenn.edu/~chinese/posguide.3rd.ch.pdf and http://www.icl.pku.edu.cn/icl_groups/corpus/spec.htm.

that the two POS tagged corpora also hold the two properties of heterogeneous annotations. The differences between the POS annotation standards are systematic. The annotations in CTB are treebank-driven, and thus consider more functional (dynamic) information of basic lexical categories. The annotations in PPD are lexicon-driven, and thus focus on more *static* properties of words. Limited to the document length, we only illustrate the annotation of verbs and nouns for better understanding of the differences.

- The CTB tag **VV** indicates common verbs that are mainly labeled as verbs (v) too according to the PPD standard. However, these words can be also tagged as nominal categories (a, vn, n). The main reason is that there are a large number of Chinese adjectives and nouns that can be realized as predicates without linking verbs.

- The tag **NN** indicates common nouns in CTB. Some of them are labeled as verbal categories (vn, v). The main reason is that a majority of Chinese *verbs* could be realized as subjects and objects without form changes.

4 Structure-based Stacking

4.1 Reducing the Approximation Error via Stacking

Each annotation data set alone can yield a predictor that can be taken as a mechanism to produce structured texts. With different training data, we can construct multiple heterogeneous systems. These systems produce similar linguistic analysis which holds the same high level linguistic principles but differ in details. A very simple idea to take advantage of heterogeneous structures is to design a predictor which can predict a more accurate target structure based on the input, the less accurate target structure and complementary structures. This idea is very close to stacked learning (Wolpert, 1992), which is well developed for ensemble learning, and successfully applied to some NLP tasks, e.g. dependency parsing (Nivre and McDonald, 2008; Torres Martins et al., 2008).

Formally speaking, our idea is to include two "levels" of processing. The first level includes one or more base predictors $f_1, ..., f_K$ that are independently built on different training data. The second level processing consists of an inference function h that takes as input $\langle \mathbf{x}, f_1(\mathbf{x}), ..., f_K(\mathbf{x}) \rangle$[3] and outputs a final prediction $h(\mathbf{x}, f_1(\mathbf{x}), ..., f_K(\mathbf{x}))$. The only difference between model ensemble and *annotation ensemble* is that the output spaces of model ensemble are the same while the output spaces of annotation ensemble are different. This framework is general and flexible, in the sense that it assumes almost nothing about the individual systems and take them as black boxes.

AS ⇒ u:44;	CD ⇒ m:134;
DEC ⇒ u:83;	DEV ⇒ u:7;
DEG ⇒ u:123;	ETC ⇒ u:9;
LB ⇒ p:1;	NT ⇒ t:98;
OD ⇒ m:41;	PU ⇒ w:552;
SP ⇒ u:1;	VC ⇒ v:32;
VE ⇒ v:13;	BA ⇒ p:2; d:1;
CS ⇒ c:3; d:1;	DT ⇒ r:15; b:1;
MSP ⇒ c:2; u:1;	PN ⇒ r:53; n:2;
CC ⇒ c:73; p:5; v:2;	M ⇒ q:101; n:11; v:1;
LC ⇒ f:51; Ng:3; v:1; u:1;	P ⇒ p:133; v:4; c:2; Vg:1;
VA ⇒ a:57; i:4; z:2; ad:1; b:1;	NR ⇒ ns:170; nr:65; j:23; nt:21; nz:7; n:2; s:1;
VV ⇒ v:382; i:5; a:3; Vg:2; vn:2; n:2; p:2; w:1;	JJ ⇒ a:43; b:13; n:3; vn:3; d:2; j:2; f:2; t:2; z:1;
AD ⇒ d:149; c:11; ad:6; z:4; a:3; v:2; n:1; r:1; m:1; f:1; t:1;	NN ⇒ n:738; vn:135; v:26; j:19; Ng:5; an:5; a:3; r:3; s:3; Ag:2; nt:2; f:2; q:2; i:1; t:1; nz:1; b:1;

Table 1: Mapping between CTB and PPD POS Tags.

4.2 A Character-based Tagger

With IOB2 representation (Ramshaw and Marcus, 1995), the problem of joint segmentation and tagging can be regarded as a character classification task. Previous work shows that the character-based approach is an effective method for Chinese lexical processing. Both of our feature- and structure-based stacking models employ base character-based taggers to generate multiple segmentation and tagging results. Our base tagger use a discriminative sequential classifier to predict the POS tag with positional information for each character. Each character can be assigned one of two possible boundary tags: "B" for a character that begins a word and "I" for a character that occurs in the middle of a word. We denote

[3] $\mathbf{x}$ is a given Chinese sentence.

a candidate character token c_i with a fixed window $c_{i-2}c_{i-1}c_ic_{i+1}c_{i+2}$. The following features are used for classification:

- Character unigrams: c_k $(i - l \leq k \leq i + l)$
- Character bigrams: c_kc_{k+1} $(i - l \leq k < i + l)$

4.3 Feature-based Stacking

Jiang et al. (2009) introduced a feature-based stacking solution for annotation ensemble. In their solution, an auxiliary tagger $\mathrm{CTag}_{\mathrm{ppd}}$ is trained on a complementary corpus, i.e. PPD, to assist the target CTB-style tagging. To refine the character-based tagger $\mathrm{CTag}_{\mathrm{ctb}}$, PPD-style character labels are directly incorporated as new features. The stacking model relies on the ability of discriminative learning method to explore informative features, which play central role to boost the tagging performance. To compare their feature-based stacking model and our structure-based model, we implement a similar system $\mathrm{CTag}_{\mathrm{ppd}\rightarrow\mathrm{ctb}}$. Apart from character uni/bigram features, the PPD-style character labels are used to derive the following features to enhance our CTB-style tagger:

- Character label unigrams: c_k^{ppd} $(i - l^{\mathrm{ppd}} \leq k \leq i + l^{\mathrm{ppd}})$
- Character label bigrams: $c_k^{\mathrm{ppd}}c_{k+1}^{\mathrm{ppd}}$ $(i - l^{\mathrm{ppd}} \leq k < i + l^{\mathrm{ppd}})$

In the above descriptions, l and l^{ppd} are the window sizes of features, which can be tuned on development data.

4.4 Structure-based Stacking

We propose a novel structured-based stacking model for the task, in which heterogeneous word structures are used not only to generate features but also to derive a sub-word structure. Our work is inspired by the stacked sub-word tagging model introduced in (Sun, 2011). Their work is motivated by the diversity of heterogeneous models, while our work is motivated by the diversity of heterogeneous annotations. The workflow of our new system is shown in Figure 1. In the first phase, one character-based CTB-style tagger ($\mathrm{CTag}_{\mathrm{ctb}}$) and one character-based PPD-style tagger ($\mathrm{CTag}_{\mathrm{ppd}}$) are respectively trained to produce heterogenous word boundaries. In the second phase, this system first combines the two segmentation and tagging results to get sub-words which maximize the agreement about word boundaries. Finally, a fine-grained sub-word tagger ($\mathrm{STag}_{\mathrm{ctb}}$) is applied to bracket sub-words into words and also to label their POS tags. We can also apply a PPD-style sub-word tagger. To compare with previous work, we specially concentrate on the *PPD-to-CTB* adaptation.

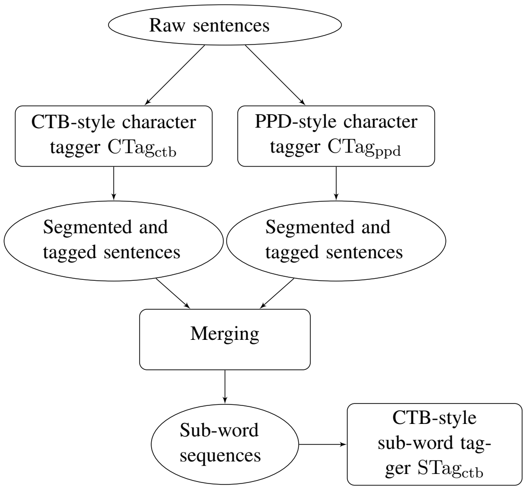
Raw sentences
CTB-style character tagger $\mathrm{CTag}_{\mathrm{ctb}}$
PPD-style character tagger $\mathrm{CTag}_{\mathrm{ppd}}$
Segmented and tagged sentences
Segmented and tagged sentences
Merging
Sub-word sequences
CTB-style sub-word tagger $\mathrm{STag}_{\mathrm{ctb}}$

Figure 1: Sub-word tagging based on heterogeneous taggers.

Following (Sun, 2011), the intermediate sub-word structures is defined to maximize the agreement of $\mathrm{CTag}_{\mathrm{ctb}}$ and $\mathrm{CTag}_{\mathrm{ppd}}$. In other words, the goal is to make merged sub-words as large as possible but not overlap with any predicted word produced by the two taggers. If the position between two continuous characters is predicted as a word boundary by any segmenter, this position is taken as a separation position of the sub-word sequence. This strategy makes sure that it is still possible to correctly *re*-segment the strings of which the boundaries are disagreed with by the heterogeneous segmenters in the sub-word tagging stage.

To train the sub-word tagger $\mathrm{STag}_{\mathrm{ctb}}$, features are formed making use of both CTB-style and PPD-style POS tags provided by the character-based taggers. In the following description, "C" refers to the content of a sub-word; "$\mathrm{T}_{\mathrm{ctb}}$" and "$\mathrm{T}_{\mathrm{ppd}}$" refers to the positional POS tags generated from $\mathrm{CTag}_{\mathrm{ctb}}$ and $\mathrm{CTag}_{\mathrm{ppd}}$; l_C, l_T^{ctb} and l_T^{ppd} are the window sizes. For convenience, we denote a sub-word with its context ...$s_{i-1}s_is_{i+1}$..., where s_i is the current token. The following features are applied:

- Unigram features: C(s_k) ($i - l_C \leq k \leq +l_C$), T$_{\mathrm{ctb}}$(s_k) ($i - l_T^{\mathrm{ctb}} \leq k \leq i + l_T^{\mathrm{ctb}}$), T$_{\mathrm{ppd}}$($s_k$) ($i - l_T^{\mathrm{ppd}} \leq k \leq i + l_T^{\mathrm{ppd}}$)
- Bigram features: C(s_k)C(s_{k+1}) ($i - l_C \leq k < i + l_C$), T$_{\mathrm{ctb}}$(s_k)T$_{\mathrm{ctb}}$(s_{k+1}) ($i - l_T^{\mathrm{ctb}} \leq k < i + l_T^{\mathrm{ctb}}$), T$_{\mathrm{ppd}}$($s_k$)T$_{\mathrm{ppd}}$($s_{k+1}$) ($i - l_T^{\mathrm{ppd}} \leq k < i + l_T^{\mathrm{ppd}}$)
- C(s_{i-1})C(s_{i+1}) (if $l_C \geq 1$), T$_{\mathrm{ctb}}$(s_{i-1})T$_{\mathrm{ctb}}$(s_{i+1}) (if $l_T^{\mathrm{ctb}} \geq 1$), T$_{\mathrm{ppd}}$($s_{i-1}$)T$_{\mathrm{ppd}}$($s_{i+1}$) (if $l_T^{\mathrm{ppd}} \geq 1$)
- Word formation features: character n-gram prefixes and suffixes for n up to 3.

Cross-validation CTag$_{\mathrm{ctb}}$ and CTag$_{\mathrm{ppd}}$ are directly trained on the original training data, i.e. the CTB and PPD data. Cross-validation technique has been proved necessary to generate the training data for sub-word tagging, since it deals with the training/test mismatch problem (Sun, 2011). To construct training data for the new heterogeneous sub-word tagger, a 10-fold *cross-validation* on the original CTB data is performed too.

5 Data-driven Annotation Conversion

It is possible to acquire high quality labeled data for a specific annotation standard by exploring existing heterogeneous corpora, since the annotations are normally highly compatible. Moreover, the exploitation of additional (pseudo) labeled data aims to reduce the estimation error and enhances a NLP system in a different way from stacking. We therefore expect the improvements are not much overlapping and the combination of them can give a further improvement.

The stacking models can be viewed as annotation converters: They take as input complementary structures and produce as output target structures. In other words, the stacking models actually learn statistical models to transform the lexical representations. We can acquire informative extra samples by processing the PPD data with our stacking models. Though the converted annotations are imperfect, they are still helpful to reduce the estimation error.

Character-based Conversion The feature-based stacking model CTag$_{\mathrm{ppd}\rightarrow\mathrm{ctb}}$ maps the input character sequence **c** and its PPD-style character label sequence to the corresponding CTB-style character label sequence. This model by itself can be taken as a corpus conversion model to transform a PPD-style analysis to a CTB-style analysis. By processing the auxiliary corpus D_{ppd} with CTag$_{\mathrm{ppd}\rightarrow\mathrm{ctb}}$, we acquire a new labeled data set $D'_{ctb} = D_{ppd\rightarrow ctb}^{CTag_{ppd\rightarrow ctb}}$. We can re-train the $CTag_{ctb}$ model with both original and converted data $D_{ctb} \cup D'_{ctb}$.

Sub-word-based Conversion Similarly, the structure-based stacking model can be also taken as a corpus conversion model. By processing the auxiliary corpus D_{ppd} with STag$_{\mathrm{ctb}}$, we acquire a new labeled data set $D''_{ctb} = D_{ppd\rightarrow ctb}^{STag_{ctb}}$. We can re-train the STag$_{\mathrm{ctb}}$ model with $D_{ctb} \cup D''_{ctb}$. If we use the gold PPD-style labels of D''_{ctb} to extract sub-words, the new model will overfit to the gold PPD-style labels, which are unavailable at test time. To avoid this training/test mismatch problem, we also employ a 10-fold cross validation procedure to add noise.

It is not a new topic to convert corpus from one formalism to another. A well known work is transforming Penn Treebank into resources for various deep linguistic processing, including LTAG (Xia, 1999), CCG (Hockenmaier and Steedman, 2007), HPSG (Miyao et al., 2004) and LFG (Cahill et al., 2002). Such work for corpus conversion mainly leverages rich sets of hand-crafted rules to convert corpora. The construction of linguistic rules is usually time-consuming and the rules are not full coverage. Compared to rule-based conversion, our statistical converters are much easier to built and empirically perform well.

6 Experiments

6.1 Setting

Previous studies on joint Chinese word segmentation and POS tagging have used the CTB in experiments. We follow this setting in this paper. We use CTB 5.0 as our main corpus and define the training, development and test sets according to (Jiang et al., 2008a; Jiang et al., 2008b; Kruengkrai et al., 2009; Zhang and Clark, 2010; Sun, 2011). Jiang et al. (2009) present a preliminary study for the annotation adaptation topic, and conduct experiments with the extra PPD data[4]. In other words, the CTB-sytle annotation is the target analysis while the PPD-style annotation is the complementary/auxiliary analysis. Our experiments for annotation ensemble follows their setting to lead to a fair comparison of our system and theirs. A CRF learning toolkit, wapiti[5] (Lavergne et al., 2010), is used to resolve sequence labeling problems. Among several parameter estimation methods provided by wapiti, our auxiliary experiments indicate that the "rprop-" method works best. Three metrics are used for evaluation: precision (P), recall (R) and balanced f-score (F) defined by 2PR/(P+R). Precision is the relative amount of correct words in the system output. Recall is the relative amount of correct words compared to the gold standard annotations. A token is considered to be correct if its boundaries match the boundaries of a word in the gold standard and their POS tags are identical.

6.2 Results of Stacking

Table 2 summarizes the segmentation and tagging performance of the baseline and different stacking models. The baseline of the character-based joint solver ($\mathrm{CTag}_{\mathrm{ctb}}$) is competitive, and achieves an f-score of 92.93. By using the character labels from a heterogeneous solver ($\mathrm{CTag}_{\mathrm{ppd}}$), which is trained on the PPD data set, the performance of this character-based system ($\mathrm{CTag}_{\mathrm{ppd}\rightarrow\mathrm{ctb}}$) is improved to 93.67. This result confirms the importance of a heterogeneous structure. Our structure-based stacking solution is effective and outperforms the feature-based stacking. By better exploiting the heterogeneous word boundary structures, our sub-word tagging model achieves an f-score of 94.03 (l_T^{ctb} and l_T^{ppd} are tuned on the development data and both set to 1).

The contribution of the auxiliary tagger is two-fold. On one hand, the heterogeneous solver provides structural information, which is the basis to construct the sub-word sequence. On the other hand, this tagger provides additional POS information, which is helpful for disambiguation. To evaluate these two contributions, we do another experiment by just using the heterogeneous word boundary structures without the POS information. The f-score of this type of sub-word tagging is 93.73. This result indicates that both the word boundary and POS information are helpful.

Devel.	P	R	F
$\mathrm{CTag}_{\mathrm{ctb}}$	93.28%	92.58%	92.93
$\mathrm{CTag}_{\mathrm{ppd}\rightarrow\mathrm{ctb}}$	93.89%	93.46%	93.67
$\mathrm{STag}_{\mathrm{ctb}}$	94.07%	93.99%	94.03

Table 2: Performance of different stacking models on the development data.

6.3 Learning Curves

We do additional experiments to evaluate the effect of heterogeneous features as the amount of PPD data is varied. Table 3 summarizes the f-score change. The feature-based model works well only when a considerable amount of heterogeneous data is available. When a small set is added, the performance is even lower than the baseline (92.93). The structure-based stacking model is more robust and obtains consistent gains regardless of the size of the complementary data.

		PPD → CTB	
#CTB	#PPD	CTag	STag
18104	7381	92.21	93.26
18104	14545	93.22	93.82
18104	21745	93.58	93.96
18104	28767	93.55	93.87
18104	35996	93.67	94.03
9052	9052	92.10	92.40

Table 3: F-scores relative to sizes of training data. Sizes (shown in column #CTB and #PPD) are numbers of sentences in each training corpus.

6.4 Results of Annotation Conversion

The stacking models can be viewed as data-driven annotation converting models. However they are not trained on "real" labeled samples. Although the target representation (CTB-style analysis in our case) is gold standard, the input representation (PPD-style analysis in our case) is labeled by a automatic tagger $\mathrm{CTag}_{\mathrm{ppd}}$. To make clear whether these stacking

[4] http://icl.pku.edu.cn/icl_res/

[5] http://wapiti.limsi.fr/

models trained with noisy inputs can *tolerate* perfect inputs, we evaluate the two stacking models on our manually converted data. The accuracies presented in Table 4 indicate that though the conversion models are learned by applying noisy data, they can refine target tagging with gold auxiliary tagging. Another interesting thing is that the gold PPD-style analysis does not help the sub-word tagging model as much as the character tagging model.

	Auto PPD	Gold PPD
$\mathrm{CTag}_{\mathrm{ppd}\rightarrow\mathrm{ctb}}$	93.69	95.19
$\mathrm{STag}_{\mathrm{ctb}}$	94.14	94.70

Table 4: F-scores with gold PPD-style tagging on the manually converted data.

6.5 Results of Re-training

Table 5 shows accuracies of re-trained models. Note that a sub-word tagger is built on character taggers, so when we re-train a sub-word system, we should consider whether or not re-training base character taggers. The error rates decrease as automatically converted data is added to the training pool, especially for the character-based tagger $\mathrm{CTag}_{\mathrm{ctb}}$. When the base CTB-style tagging is improved, the final tagging is improved in the end. The re-training does not help the sub-word tagging much; the improvement is very modest.

$CTag_{ctb}$	$STag_{ctb}$	P(%)	R(%)	F
$D_{ctb} \cup D'_{ctb}$	- -	94.46	94.06	94.26
$D_{ctb} \cup D'_{ctb}$	D_{ctb}	94.61	94.43	94.52
D_{ctb}	$D_{ctb} \cup D''_{ctb}$	94.05	94.08	94.06
$D_{ctb} \cup D'_{ctb}$	$D_{ctb} \cup D''_{ctb}$	94.71	94.53	94.62

Table 5: Performance of re-trained models on the development data.

6.6 Comparison to the State-of-the-Art

Table 6 summarizes the tagging performance of different systems. The baseline of the character-based tagger is competitive, and achieve an f-score of 93.41. By better using the heterogeneous word boundary structures, our sub-word tagging model achieves an f-score of 94.36. Both character and sub-word tagging model can be enhanced with automatically converted corpus. With the pseudo labeled data, the performance goes up to 94.11 and 94.68. These results are also better than the best published result on the same data set that is reported in (Jiang et al., 2009).

Test	P	R	F
(Sun, 2011)	- -	- -	94.02
(Jiang et al., 2009)	- -	- -	94.02
(Wang et al., 2011)	- -	- -	94.18[6]
Character model	93.31%	93.51%	93.41
+Re-training	93.93%	94.29%	94.11
Sub-word model	94.10%	94.62%	94.36
+Re-training	**94.42%**	**94.93%**	**94.68**

Table 6: Performance of different systems on the test data.

7 Conclusion

Our theoretical and empirical analysis of two representative popular corpora highlights two essential characteristics of heterogeneous annotations which are explored to reduce approximation and estimation errors for Chinese word segmentation and POS tagging. We employ stacking models to incorporate features derived from heterogeneous analysis and apply them to convert heterogeneous labeled data for re-training. The appropriate application of heterogeneous annotations leads to a significant improvement (a relative error reduction of 11%) over the best performance for this task. Although our discussion is for a specific task, the key idea to leverage heterogeneous annotations to reduce the approximation error with stacking models and the estimation error with automatically converted corpora is very general and applicable to other NLP tasks.

Acknowledgement

This work is mainly finished when the first author was in Saarland University and DFKI. At that time, this author was funded by DFKI and German Academic Exchange Service (DAAD). While working in Peking University, both author are supported by NSFC (61170166) and National High-Tech R&D Program (2012AA011101).

[6] This result is achieved with much unlabeled data, which is different from our setting.

References

Aoife Cahill, Mairead Mccarthy, Josef Van Genabith, and Andy Way. 2002. Automatic annotation of the penn treebank with lfg f-structure information. In *Proceedings of the LREC Workshop on Linguistic Knowledge Acquisition and Representation: Bootstrapping Annotated Language Data, Las Palmas, Canary Islands*, pages 8–15.

Julia Hockenmaier and Mark Steedman. 2007. Ccgbank: A corpus of ccg derivations and dependency structures extracted from the penn treebank. *Computational Linguistics*, 33(3):355–396.

Wenbin Jiang, Liang Huang, Qun Liu, and Yajuan Lü. 2008a. A cascaded linear model for joint Chinese word segmentation and part-of-speech tagging. In *Proceedings of ACL-08: HLT*, pages 897–904, Columbus, Ohio, June. Association for Computational Linguistics.

Wenbin Jiang, Haitao Mi, and Qun Liu. 2008b. Word lattice reranking for Chinese word segmentation and part-of-speech tagging. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 385–392, Manchester, UK, August. Coling 2008 Organizing Committee.

Wenbin Jiang, Liang Huang, and Qun Liu. 2009. Automatic adaptation of annotation standards: Chinese word segmentation and pos tagging – a case study. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 522–530, Suntec, Singapore, August. Association for Computational Linguistics.

Canasai Kruengkrai, Kiyotaka Uchimoto, Jun'ichi Kazama, Yiou Wang, Kentaro Torisawa, and Hitoshi Isahara. 2009. An error-driven word-character hybrid model for joint Chinese word segmentation and pos tagging. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 513–521, Suntec, Singapore, August. Association for Computational Linguistics.

Thomas Lavergne, Olivier Cappé, and François Yvon. 2010. Practical very large scale CRFs. pages 504–513, July.

Yusuke Miyao, Takashi Ninomiya, and Jun ichi Tsujii. 2004. Corpus-oriented grammar development for acquiring a head-driven phrase structure grammar from the penn treebank. In *IJCNLP*, pages 684–693.

Hwee Tou Ng and Jin Kiat Low. 2004. Chinese part-of-speech tagging: One-at-a-time or all-at-once? word-based or character-based? In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 277–284, Barcelona, Spain, July. Association for Computational Linguistics.

Zheng-Yu Niu, Haifeng Wang, and Hua Wu. 2009. Exploiting heterogeneous treebanks for parsing. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 46–54, Suntec, Singapore, August. Association for Computational Linguistics.

Joakim Nivre and Ryan McDonald. 2008. Integrating graph-based and transition-based dependency parsers. In *Proceedings of ACL-08: HLT*, pages 950–958, Columbus, Ohio, June. Association for Computational Linguistics.

Lance Ramshaw and Mitch Marcus. 1995. Text chunking using transformation-based learning. In David Yarowsky and Kenneth Church, editors, *Proceedings of the Third Workshop on Very Large Corpora*, pages 82–94, Somerset, New Jersey. Association for Computational Linguistics.

Weiwei Sun and Jia Xu. 2011. Enhancing Chinese word segmentation using unlabeled data. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 970–979, Edinburgh, Scotland, UK., July. Association for Computational Linguistics.

Weiwei Sun, Rui Wang, and Yi Zhang. 2010. Discriminative parse reranking for Chinese with homogeneous and heterogeneous annotations. In *Proceedings of Joint Conference on Chinese Language Processing (CIPS-SIGHAN)*, Beijing, China, August.

Weiwei Sun. 2010. Word-based and character-based word segmentation models: Comparison and combination. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1211–1219, Beijing, China, August. Coling 2010 Organizing Committee.

Weiwei Sun. 2011. A stacked sub-word model for joint Chinese word segmentation and part-of-speech tagging. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1385–1394, Portland, Oregon, USA, June. Association for Computational Linguistics.

André Filipe Torres Martins, Dipanjan Das, Noah A. Smith, and Eric P. Xing. 2008. Stacking dependency parsers. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 157–166, Honolulu, Hawaii, October. Association for Computational Linguistics.

Yiou Wang, Jun'ichi Kazama, Yoshimasa Tsuruoka, Wenliang Chen, Yujie Zhang, and Kentaro Torisawa. 2011. Improving chinese word segmentation and pos tagging with semi-supervised methods using large auto-analyzed data. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 309–317, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

David H. Wolpert. 1992. Original contribution: Stacked generalization. *Neural Netw.*, 5:241–259, February.

Fei Xia. 1999. Extracting tree adjoining grammars from bracketed corpora. In *Proceedings of Natural Language Processing Pacific Rim Symposium*, pages 398–403.

Yue Zhang and Stephen Clark. 2008. Joint word segmentation and POS tagging using a single perceptron. In *Proceedings of ACL-08: HLT*, pages 888–896, Columbus, Ohio, June. Association for Computational Linguistics.

Yue Zhang and Stephen Clark. 2010. A fast decoder for joint word segmentation and POS-tagging using a single discriminative model. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 843–852, Cambridge, MA, October. Association for Computational Linguistics.