Using Collaborative Training Method to build Vietnamese Dependency Treebank

Guoke Qiu[1*], Jianyi Guo[1], Zhengtao Yu[2], Yantuan Xian[2], Cunli Mao[2]

(1.The School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, Yunnan 650500，China
2. The Key Laboratory of Intelligent Information Processing, Kunming University of Science and Technology, Kunming, Yunnan 650500，China)

Abstract: For the difficulty of marking Vietnamese dependency tree, this paper proposed the method which combined MST algorithm and improved Nivre algorithm to build Vietnamese dependency treebank. The method took full advantage of the characteristics of collaborative training. Firstly, we built a bit samples. Secondly, we used the samples to build two weak learners with two fully redundant views. Then, we marked a large number of unmarked samples mutually. Next, we selected the samples of high trust to relearn and built a dependency parsing system. Finally, we used 5000 Vietnamese sentences marked manually to do tenfold cross-test and obtained the accuracy of 76.33%. Experimental results showed that the proposed method in this paper could take full advantage of unmarked corpus to effectively improve the quality of dependency treebank.

Keywords: Dependency Treebank; Vietnamese; Collaborative Training; Dependency Parsing;

1 Introduction

Vietnam is a close neighbour of China. The mutual translation will play an important role for exchanges between the two countries. A large-scale dependency treebank can provide strong support for machine translation and other upper applications. Therefore, building Vietnamese dependency treebank has an important practical significance. Currently, the construction of dependency treebank for English and other large languages has got some achievements. But research about Vietnamese is still relatively less and there are a lack of large-scale Vietnamese dependency treebanks.

Currently, in the field of Vietnamese information processing, there are some research achievements in morphology and bilingual alignment method[1,2,3]. But research on building dependency treebank is relatively inadequate. With the rapid development of statistical learning, today more and more researchers use this method to study language information processing. Among them, Lai and others used the idea of span and statistical learning to solve the problem of Chinese dependency parsing in 2001[4]. Yamada and others converted the English sentences in the Penn Treebank to the dependency structure completely in 2003. Then they used the statistical learning method to analyze these sentences and finally achieved the accuracy of 90.3%[5]. Ma Jinshan built the SVM dependency parsing model through using the marked Chinese dependency treebank in 2004 and finally solved the Chinese dependency parsing[6]. These methods above mainly relied on supervised learning of dependency treebank resource to achieve dependency parsing. P.T.Nguyen and others converted ten thousand phrase trees in the Penn Treebank to

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61262041, 61363044 and 61472168) and the key project of National Natural Science Foundation of Yunnan province (Grant No. 2013FA030).

dependency trees in 2013[7]. But the scale was still relatively small.

The foundation of dependency parsing is the construction of dependency treebank. However, marking dependency treebank is very difficult and currently there isn't a mature dependency parser. For the construction of Vietnamese dependency treebank, marking it manually is very difficult and this process requires a lot of manpower and other material resource. Moreover, in reality there is a lot of unmarked crude corpus and the corpus has not undergone any processing. Therefore, how to use the corpus to build Vietnamese dependency treebank effectively has become an important issue for Vietnamese dependency parsing.

Based on the features of Vietnamese, this paper proposed the method which combined the maximum spanning tree (MST) algorithm with the improved Nivre algorithm to build Vietnamese dependency treebank. The method was aimed to explore how to use unmarked crude corpus effectively with the help of collaborative training. Firstly, we selected some Vietnamese sentences marked manually as the initial training corpus and used them to build two weak learners. Secondly, we used a large number of unmarked Vietnamese sentences to mark each other and extracted a sample of high trust to train and update on the two learners repeatedly. Finally, we achieved building a Vietnamese dependency treebank of high accuracy successfully. Experimental results showed that the proposed method in this paper improved the UAS, LAS, RA and the accuracy of other aspects more significantly than other methods.

2 Related Theories

2.1 The Maximum Spanning Tree (MST) Algorithm

Dependency-tree parsing as the search for the maximum spanning tree (MST) in a graph was proposed by McDonald et al.(2005c). This formulation leads to efficient parsing algorithms for both projective and non-projective dependency trees with the Eisner algorithm(Eisner, 1996) and the Chu-Liu-Edmonds algorithm(Chu and Liu, 1965; Edmonds, 1967) respectively. The formulation works by defining the score of a dependency tree to be the sum of edge scores:

$$s(x, y) = \sum(i,j) \in y$$
$$s(i,j)$$

where x = x1 ···xn is an input sentence and y a dependency tree for x. We can view y as a set of tree edges and write (i,j) ∈ y to indicate an edge in y from word xi to word xj.

We call this first-order dependency parsing since scores are restricted to a single edge in the dependency tree. The score of an edge is in turn computed as the inner product of a high-dimensional feature representation of the edge with a corresponding weight vector:

$$s(i,j) = \mathbf{w} \cdot \mathbf{f}(i,j)$$

This is a standard linear classifier in which the weight vector w are the parameters to be learned during training. We should note that **f**(i,j) can be based on arbitrary features of the edge and the input sequence x.

Given a directed graph G = (V,E), the maximum spanning tree (MST) problem is to find the highest scoring subgraph of G that satisfies the tree constraint over the vertices V . By defining a graph in which the words in a sentence are the vertices and there is a directed edge between all words with a score as calculated above, McDonald et al. (2005c) showed that dependency parsing is equivalent to finding the MST in this graph. Furthermore, it was shown that this formulation can lead to state-of-the-art results when combined with discriminative learning algorithms. Although the MST formulation applies to any directed graph, our feature representations and one of the parsing algorithms (Eisner's) rely on a linear ordering of the vertices, namely the order of the words in the sentence.

In this paper, we expressed the dependency tree of a Vietnamese sentence $S=\{s_1,s_2,...,s_n\}$ as a directed graph $G=(V,E)$, where the words in the sentence constituted a set of vertexs of G and $E\subseteq[1:n]\times[1:n]$ represented the dependency. If there was a directed connection from vertex i to vertex j in the dependency tree, there was a directed edge between i and j. The weight of each directed edge was defined as $score(i,j,y)$, which represented the probability of j depending on I and y was a dependency type. The weight of a dependency tree was the sum of the weights of all directed edges. Therefore, this dependency parsing method would convert looking for the best result into searching for the maximum spanning tree in the directed graph $G=(V,E)$：

$$T=\underset{G=(\mathrm{V},\mathrm{E})}{\arg\max}\sum_{(\mathrm{i},\mathrm{j})\in\mathrm{E}} score(\mathrm{i},\mathrm{j},\mathrm{y})$$

2.2 The Improved Nivre Algorithm

The Nivre algorithm is based on the process of state transition. The algorithm can obtain the model of dependency parsing through training. The model can predict the next state according to the current state and the features of input sentences and previous decisions. During the dependency parsing, the analyzer transfers greedily from a primitive state to a subsequent state according to the forecast sets of the model until it reaches the end state.

For the deterministic Nivre algorithm, the division about the Reduce operation and the Shift operation is not very accurate. In order to solve this problem, the paper proposed an improved Nivre algorithm.

In the Nivre algorithm, a parser can be expressed as a triad <S,I,A>, Where S and I are stacks. The input sequence to be parsed is stored in I. A is a set and it can be used to store the determinate dependency items in the process of parsing. Given an input sequence Sen, the parser is firstly initialized as $< nil, Sen, \varnothing >$ .The parser analyzes the dependency between the top element t of stack S and the top element n of stack I. Then, the parser takes appropriate action to move the elements and control the algorithm iteration until stack I is empty. At the moment, the parser stops iterativing and outputs the dependency sequences of set A. The Nivre algorithm defines a total of four operations:

（1） Right. If t depends on n in the current triad $<t\,|\,S,n\,|\,I,A>$, the item t->n is added into set A and the top element t of stack S is popped up. Finally, the triad becomes $<S,n\,|\,I,A\cup\{(t\rightarrow n)\}>$.

（2） Left. If n depends on t in the current triad $<t|S,n|I,A>$, the item n->t is added into set A and the element n is pushed into stack S. Finally, the triad becomes $<n|t|S,I,A\cup\{(n\to t)\}>$.

If there isn’t any dependency between n and t, the improved Nivre algorithm makes a clear definition about the Reduce operation and the Shift operation.

（3） Reduce. If there isn’t any dependency between n and t, t has a parent node in its left side and there is a dependency between its parent node and n, the parser pops up t from stack S. Finally, the triad becomes $<S,n|I,A>$.

（4） Shift. When the Right, Left and Reduce can’t be met, n is pushed into stack S. Finally, the triad becomes $<n|t|S,I,A>$.

2.3 The Bottom-Up Algorithm and the Top-Down Algorithm

The Bottom-Up Computation (BUC) algorithm is a bottom-up approach (Beyer and Ramakrishnan 1999). BUC processes the partitions starting from a single attribute and moves towards the apex of the lattice. BUC relies on APRIORI-like pruning to reduce the computation space. BUC is a divide and conquer strategy, and partitioning is its major cost. BUC can be used to compute either a full data cube or an iceberg cube. Due to its pruning power, BUC works especially well at computing iceberg cubes for sparse database tables. As well, BUC is not memory intensive. When the database is dense, the dividing into partitions costs more and the pruning is less effective, so the overall performance of BUC degrades. According to extensive studies, BUC is faster than TDC in most cases (Findlater and Hamilton 2003).

As the name implies, the Top-Down Computation (TDC) algorithm (Findlater and Hamilton 2003) is a top-down approach that starts from the least aggregated group-bys at the top of the lattice and works its way down to the most aggregated group-bys at the bottom. Each underlined group-by is also an ordering, i.e., a child group-by that permits the shared computation of its parent and other ancestor group-bys during the pass in which it is being computed. Using orderings, the number of passes over the database can be reduced. TDC uses orderings to cover all group-bys. To cover the 2^m group-bys of an mdimensional data cube, 2^{m-1} orderings are required (Findlater and Hamilton 2003). When processing an ordering, TDC simultaneously aggregates all group-bys that are prefixes of it. Shared computation is the main advantage of TDC. The main disadvantage of TDC is weak pruning, i.e., it is relatively poor at identifying cases where pruning is possible.

2.4 Basic principle of Collaborative Training

Cooperative training was proposed by A.Blum and T.Mitchell in 1998. This method assumes that the data set has two fully redundant views, that is to say the data set has several attribute sets to meet the following two conditions: Firstly, the training data of each attribute set is enough to describe the problem and each attribute set can obtain a weak learner through learning. Secondly, any two attribute sets are independent conditionally in the process of marking.

Firstly, cooperative training requires some marked samples to train their classifiers in both views respectively. Secondly, each classifier selects some samples of high confidence from unmarked samples. Next, the selected samples are added into another classifier to train after being marked, so that the classifier can use the newly added marked samples to train and update. Finally, the two classifiers update and iterate constantly until the parameters of the model converge. Research has shown that when the assumption about fully redundant views is established, cooperative training can effectively use unmarked samples to improve training performance. The standard collaborative training method is shown in figure 1, where X1 and X2 respectively represent the corresponding samples of view 1 and view 2.

```
input:     the data set L about marked samples
           the data set U about unmarked samples
process:   randomly select u samples as U ' from U
             K iterations
             use L and X1 to train the learner h1
             use L and X2 to train the learner h2
             use h1 to mark P positive samples and n negative samples of U '
             use h2 to mark P positive samples and n negative samples of U '
             add the samples marked in the previous step into L
             randomly select 2P + 2n samples from U and add them into U '
```

Figure 1 the standard collaborative training method

3 Related Work

3.1 Selecting and processing of corpus

Corpus is a very important concept in the field of natural language processing. The selection of corpus is very important for the construction of treebank. Because corpus is important for both annotation and experiment.

In this paper, the corpus originally came from some crude news corpus crawled from the Radio The Voice of Vietnam(Abbreviation:VOV). The news corpus was covered with politics, economy, military, sports, entertainment and other aspects, thus ensuring a diversity of experimental data. The next step was to process the original crude data manually to obtain 100,000 standard Vietnamese sentences. Then 10,000 Vietnamese sentences among them were selected to do manual annotation and repeated proofreader in order to obtain the initial training corpus and experimental test corpus. Either of them was a small Vietnamese dependency treebank and they both contained 5000 marked Vietnamese sentences. The remaining 90,000 unmarked Vietnamese sentences were used as the experimental extended corpus. The selecting and processing of corpus was shown in figure 2.

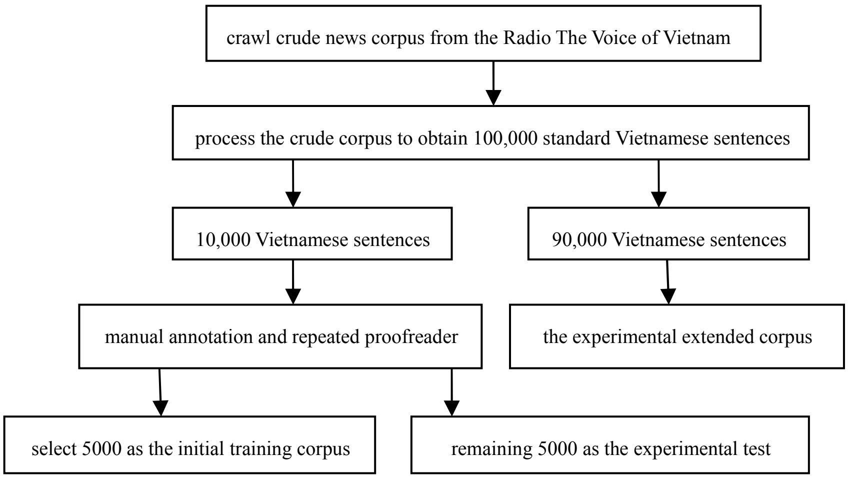
crawl crude news corpus from the Radio The Voice of Vietnam
process the crude corpus to obtain 100,000 standard Vietnamese sentences
10,000 Vietnamese sentences
90,000 Vietnamese sentences
manual annotation and repeated proofreader
the experimental extended corpus
select 5000 as the initial training corpus
remaining 5000 as the experimental test

Figure 2 the selecting and processing of corpus

3.2 Marking Vietnamese dependency

The formulation of marking standard is not only the first step to build treebank but also one of the most important work. The marking standard of high quality should be able to accurately reflect the inherent regular pattern of language and lay a good foundation for the next research. Meanwhile, in order to facilitate users to understand the marking standard, it is also crucial of improving the marking quality and manual proofreader efficiency. In addition, the appropriate standard will play a positive role for training and testing data.

Through analyzing Vietnamese grammar, firstly we must develop a dependency marking standard table in line with the features of Vietnamese. The marking standard should contain two elements: the first is that which words will exist dependency in a Vietnamese sentence; the second is how to define their dependency types.

For question one: the paper considered the first element from the semantic point. In a sentence, there should be a dependency between words which have some relationship in semantic level, that is to say generating dependency between them can promote new semantic. The paper called it the semantic principle. When marking dependency, the semantic principle should be given the priority.

Example one: As shown in figure 3, in the Vietnamese sentence**“ Cô là（she）một (is) xinh đẹp(beautiful) cô gái(girl)”,** the two words **“một (is) and cô gái (girl)”** generate relationship can just promote new semantic. So there is a dependency between the two words.

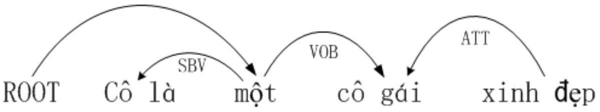

Figure 3 a dependency marking example

Secondly, in a sentence, some words play a leading role for the expression of the sentence. They are essential in the sentence, so they are key words. However, some words play a auxiliary role in the sentence, only modifying the key words, even removing them will not affect the expression of the sentence, so they are minor words. When marking dependency, it should be ensure that the key words must be located in the core of the dependency. The minor words should depend on the key words. The paper called it the trunk principle. So that it is easy to extract the main components of a sentence in subsequent applications through the dependency. Vietnamese dependency parsing is mainly led by the predicate and analyzes the relationship between the predicate and other components.

Example two: In the Vietnamese sentence **"cảm_ơn (thank) các_anh_chị (you) của (is) thịnh_tình (heart) tiếp_đãi (hospitality)"**, the words **"cảm_ơn (thank) and tiếp_đãi (hospitality)"** are the key words of the sentence. Other words only modify them. So, there should be a dependency between the two words.

For question two: the paper defined their dependency type, that is to say the paper built a Vietnamese dependency marking standard table. In order to cover various grammatical phenomena more accurately, but not lead to the problem of marking difficulty and sparse data because of excessive dependency types, the paper developed 14 kinds of dependency marking standard in line with the features of Vietnamese through analyzing Vietnamese grammar. They were shown in table 1.

Table 1 Vietnamese dependency marking standard table

Dependency Type	Tag	Description	Example
subject-predicate	SBV	subject-verb	I give him an apple (I <- to)
verb-object	VOB	verb-object	I gie him an apple (to -> apple)
indirect-object	IOB	indirect-object	I give him an apple (to -> him)
fronting-object	FOB	fronting-object	He eats any fruit (fruit <- eat)
double	DBL	double	My mother called me to eat dinner (called -> I)
attribute	ATT	attribute	little poplar (small <- poplar)
adverbial	ADV	adverbial	very quick (very <- quick)
complement	CMP	complement	end to eat dinner (eat -> end)
coordinate	COO	coordinate	tree and grass (tree -> grass)
preposition-object	POB	preposition-object	in the room (in -> inside)
left adjunct	LAD	left adjunct	tree and grass (and <- grass)
right adjunct	RAD	right adjunct	students (students -> s)
independent structure	IS	independent structure	two sentences are structurally independent
head	ROOT	head	the core of the whole sentence

3.3 Construction of initial corpus

According to the above table, the paper marked 5000 Vietnamese sentences manually as the initial corpus. The annotation storage and structure of the dependencies was shown in table 2.

Table 2 annotation storage and structure of dependencies of the initial corpus

Vietnamese word	part-of-speech	the location of dependency node	dependency
Thủy	N	2	SBV
thủ	V	0	ROOT
tàu	N	2	FOB
sẽ	R	5	ADV
giao	V	2	VOB
lưu	V	5	VOB
thể	R	6	ADV
thao	N	6	DOB
…	…	…	…

3.4 Feature Selection

After a deep research about Vietnamese, it is found that the structure of Vietnamese is relatively simple. Therefore, the paper selected the current word W0, its previous word W-1, its front second word W-2, the next word W1, the next second word W2 and the part-of-speech of the current word POS0, the part-of-speech of its previous word POS-1, the part-of-speech of its front second word POS-2, the part-of-speech of the next word POS1, the part-of-speech of the next second word POS2 as the features. The feature selection was shown in table 3.

Table 3 Feature Selection

1	W_n	word in different locations: n=-2,-1,0,1,2
2	POS_n	part-of-speech in different locations: n=-2,-l,0,l,2

The selection way fully took into account the features of Vietnamese and it had a better coverage on them. Because it not only met the basic needs but also effectively avoided the sparse data due to excessive feature selection.

4 Using Collaborative Training to build Vietnamese Dependency Treebank

The main idea of building Vietnamese dependency treebank based on collaborative training is to build two learner models and do collaborative learning. This paper proposed the method which combined the MST algorithm[8] and the improved Nivre algorithm[9] to build two weak learner models. In the process of collaborative training, this paper used the K-Best algorithm to select one learner's forecast results, and regarded the results of high confidence as the input of the other learner to train and update repeatedly until the parameters of the learner models converged.

4.1 Confidence Judgment Criterion

After having the initial training corpus, the next problem to be considered was how to use a large number of unmarked samples effectively for collaborative training. In the process of predicting unmarked samples, the confidence judgment criterion was particularly important. In order to measure the forecast results, we used the K-Best algorithm to determine the confidence. If the K weight scores of forecast results were closer, it showed that the confidence was lower. If the weight difference of forecast results was greater, the results were more accurate. Then we chose the forecast result of the highest weight score as the marked result of Vietnamese sentence.

This paper used the following three methods to calculate confidence:

Method one: the score difference sum's reciprocal of any two different results in K-Best results:

$$H = \frac{1}{\sum_{i=1}^{k-1}\sum_{j=i+1}^{k}(score_i - score_j)} \quad \text{（Formula 1）}$$

where $score_i$ and $score_j$ were the scores of weak learners to the i-th and j-th sentences' forecast results.

Method two: the growth rate' reciprocal of 1-Best with respect to 2-Best in K-Best results:

$$H = \frac{score_2}{score_1 - score_2} \quad \text{（Formula 2）}$$

Method three: the entropy of K results:

$$H = \sum_{i=1}^{k} -p_i \log p_i \quad \text{（Formula 3）}$$

$$p_i = \frac{score_i}{\sum_{j=1}^{k} score_j}$$

Method one and Method two showed that the difference of forecast results was greater, the confidence was more higher. Method three used the entropy to determine the confidence.

4.2 Collaborative Training combining MST Algorithm with Improved Nivre Algorithm

Models of MST and Nivre are both data-driven models. So McDonald and Nivre proposed a combination method[10] which regarded the forecast results of one model as the training corpus of the other to promote their mutual learning of the two models.

Firstly, the paper used some of the initial corpus samples marked previously to obtain two weak dependency parsing learners S1and S2 through training as two fully redundant views. The learner S1 was based on the MST algorithm. The learner S2 was based on the improved Nivre algorithm. Secondly, the paper randomly selected some of unmarked Vietnamese sentences as set A and set B from many unmarked samples. Then the paper respectively used the set A and the set B to predict Vietnamese dependency. The paper regareded 100 unmarked Vietnamese sentences as a unit and used the learner S1 to predict the 100 sentences. Next, the paper used the formula 1 to select 20 sentences of high confidence to mark, and then added these marked sentences into the learner S2 to train and update. Conversely, the paper also regareded 100 unmarked Vietnamese sentences as a unit and used the learner S2 to predict these 100 sentences. Next, the paper used the formula 1 to select 20 sentences of high confidence to mark, and then added these marked sentences into the learner S1 to train and update. This cycle repeated until the parameters of the learner S1 and the learner S2 became unchanged. The process of collaborative training was shown in figure 4.

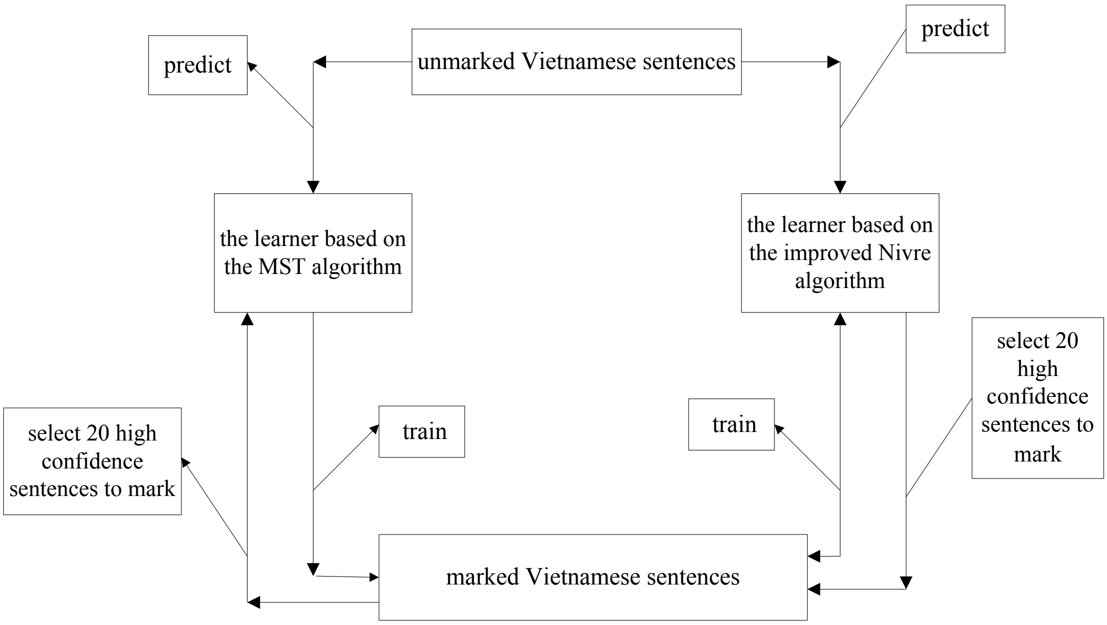
predict
unmarked Vietnamese sentences
predict
the learner based on the MST algorithm
the learner based on the improved Nivre algorithm
select 20 high confidence sentences to mark
train
train
select 20 high confidence sentences to mark
marked Vietnamese sentences

Figure 4 the process of collaborative training

After the parameters of the training model converged, the paper used the two learners to do Vietnamese dependency parsing for a large number of unmarked Vietnamese sentences and build Vietnamese dependency treebank. In the process of building Vietnamese dependency treebank, if the forecast results of the two learners were consistent, the results were correct. If the results were inconsistent, the paper used the formula 2 and the formula 3 to calculate confidence.

Finally, the paper used the formula 1, the formula 2 and the formula 3 to respectively calculate their average score for the forecast results of the two learners and the paper selected a higher score as the correct prediction. The paper used the model to do Vietnamese dependency parsing for 90,000 unmarked Vietnamese sentences. Then the paper also used the model to do Vietnamese dependency parsing for 5000 Vietnamese sentences in the test corpus and ultimately build a large-scale Vietnamese dependency treebank.

5 Experimental Results and Analysis

5.1 Experimental Evaluation Method

In the experiments, the paper used the Unlabeled Attachment Score(UAS), the Labeled Attachment Score(LAS) and the Root Accuracy(RA) as the evaluation standard of the final built dependency treebank. They were defined as follows:

$$UAS = \frac{\text{the number of words whose arcs are correct}}{\text{the number of all words}} * 100\%$$

$$LAS = \frac{\text{the number of words whose dependency arcs and dependencies are both correct}}{\text{the number of all words}} * 100\%$$

$$RA = \frac{\text{the number of sentences whose roots are correct}}{\text{the number of all sentences}} * 100\%$$

5.2 Experimental Design

In order to prove that the collaborative training can use a large number of unmarked Vietnamese sentences corpus effectively and improve the accuracy of dependency treebank, the paper designed three groups of comparative experiments to respectively build Vietnamese dependency treebank and compared the experimental results of different methods.

The first comparative experiment: Firstly, the paper used the three algorithms based on the maximum entropy to design experiments to build Vietnamese dependency treebank. They were respectively the Bottom-Up algorithm, the Top-Down algorithm and the MST algorithm. It was easy to find that the MST algorithm in USA, LAS and RA was the highest after the comparison of the experimental results. The first comparative experimental results were shown in table 4.

The second comparative experiment: Secondly, the paper used the improved Nivre algorithm to design experiments to build Vietnamese dependency treebank. Its experimental result in USA, LAS and RA was higher than that of the MST algorithm. Therefore, in order to take full advantage of these two algorithms and enhance their complementarity, the paper used the combination of the MST algorithm and the improved Nivre algorithm through collaborative training to build Vietnamese dependency treebank.

The third comparative experiment: Finally, the paper used the collaborative training to design experiments to build Vietnamese dependency treebank. In this experiment, the paper expanded 90000 Vietnamese sentences corpus. It was easy to find that the UAS, LAS and RA of the dependency treebank based on collaborative training had been significantly improved compared with the improved Nivre algorithm after the comparison of the experimental results. In addition, we compared the collaborative training method with the latest Chinese-Vietnamese bilingual-word-alignment-corpus-based method. It was easy to find that the UAS, LAS and RA of the dependency treebank based on the former were all higher than the latter. It fully proved the effectiveness of the collaborative training method. The second and third comparative experimental results were shown in table 5.

Table 4 The first comparative experimental results

method	training corpus	extended unmarked corpus	test corpus	UAS%	LAS%	RA%
the Bottom-Up algorithm	5000	90000	5000	70.62	67.32	75.45
the Top-Down algorithm	5000	90000	5000	72.25	68.30	77.35
the MST algorithm	5000	90000	5000	75.25	71.01	79.68

Table 5 The second and third comparative experimental results

method	training corpus	extended unmarked corpus	test corpus	UAS%	LAS%	RA%
the MST algorithm	5000	90000	5000	75.25	71.01	79.68
the improved Nivre algorithm	5000	90000	5000	78.35	72.32	80.76
the latest Chinese-Vietnamese bilingual-word-alignment-corpus-based method	5000	90000	5000	78.93	74.22	83.32
the collaborative training method	5000	90000	5000	80.36	76.33	83.56

5.3 Experimental Results Analysis

After careful analysis of the experimental results, it was easy to find that the accuracy of Vietnamese dependency treebank based on the collaborative training method in UAS, LAS and RA was the highest. Because the MST algorithm uses the dependency tree of the whole sentence for training and utilizes the maximum spanning tree to search for the optimal dependency tree in the process of building dependency treebank. The intermediate results of parsing cannot be applied to the subsequent analysis, leading to the low accuracy. However, the improved Nivre algorithm is based on state transition process for training and it searches for the partial optimum transfer status until the whole sentence parsing ends in the process of building dependency treebank. So the improved Nivre algorithm has the features of locality and greed and this is the reason why it has low accuracy.

However, the collaborative training method the paper proposed makes full use of the complementarity of the MST algorithm and the improved Nivre algorithm. It regards the forecast results of one model as the input of the other. When the analysis accuracy of the two models differs little, the combined model can improve the accuracy of UAS, LAS and RA significantly. In this paper, the final built dependency treebank contained 100,000 Vietnamese sentences and it eventually obtained the accuracy of 76.33%. Compared with other methods to build Vietnamese dependency treebank, the final built dependency treebank in this paper had a larger scale and higher accuracy.

Because of the complex language features, more grammar rules and types of dependency in Vietnamese, there would be inevitably some errors in the final built treebank. Although these errors were less, they were difficult to find. So the repeated manual correction for the final built dependency treebank was necessary and this work had a great significance for improving the quality of the final built dependency treebank.

6 Conclusion and Future Work

In case that the sample corpus was relatively less, we studied the three algorithms based on the maximum entropy and the improved Nivre algorithm in depth and found that only using one algorithm to build Vietnamese dependency treebank was not very satisfactory. Therefore, the paper proposed the method which combined the MST algorithm and the improved Nivre algorithm to build Vietnamese dependency treebank. The method was based on the idea of collaborative training. Experimental results showed that the proposed method had a better effect than only using an algorithm. It also had a stable parsing performance and could effectively improve the accuracy of the final built dependency treebank. The Vietnamese dependency treebank resource was relatively inadequate. But the proposed method in this paper could effectively use unmarked Vietnamese sentences to build Vietnamese dependency treebank. The method solved the experimental difficulty due to the lack of sample corpus. At the same time, the method avoided the process of manually marking Vietnamese dependency treebank and fully saved the time of manpower and other material resource.

After a detailed study, it is not difficult to find such a grammatical phenomenon in Vietnamese. In some Vietnamese sentences, some words represent a development trend of things. But these words doesn’t make much sense for the expression of the whole sentence. Moreover, there isn’t any dependency between these words and other ingredients of the sentence. For example, there is a Vietnamese sentence **“Hoa (Flower) đang (is) dần (slowly) dần (slowly) nở (open) ra.”**. In the sentence, **“ra”** is such a word. It represents a development trend that the flower is slowly open. In this case, the parsed dependencies based on the proposed method in this paper must be wrong. Moreover, this type of error is difficult to find through manual correction. So in the future work, we will do further research about how to remove these wrong dependencies in the process of building dependency treebank. If these dependencies are removed, the accuracy of the dependency treebank can also be significantly improved.

In addition, we will combine more methods to build Vietnamese dependency treebank through collaborative training and compare these methods with the proposed method in this paper. Our ultimate goal is to build a more-fusion-method, higher-accuracy and more-larger-scale Vietnamese dependency treebank.

References

[1].Le-Hong, P., and TMH. Nguyen, "Part-of-Speech Induction for Vietnamese", The Fifth International Conference on Knowledge and Systems Engineering (KSE 2013) , vol. 2, Hanoi, Vietnam, Springer-Verlag, pp. 261-272, 10/2013.

[2].Le-Hong, P., T M H. Nguyen, M. Rossignol, and A. Roussanaly, " An empirical study of maximum entropy approach for part-of-speech tagging of Vietnamese texts", Actes du Traitement Automatique des Langues Naturelles (TALN-2010), Montreal, Canada, 2010.

[3]Dinh, Q T., T M H. Nguyen, X L. Vu, M. Rossignol, P. Le-Hong, and C T. Nguyen, "Word segmentation of Vietnamese texts: a comparison of approaches", Proceedings of The Sixth International Conference on Language Resources and Evaluation, Marrakech, Morocco, 2008.

[4].T.B.Y. Lai, C.N. Huang, M. Zhou, J.B. Miao, and K.C.Siu. Span-based Statistical Dependency Parsing of Chinese. Proc. NLPRS. 2001: 677-684

[5].H. Yamada and Y. Matsumoto. Statistical Dependency Analysis with Support Vector Machines. Proc. of the 8th Intern. Workshop on Parsing Technologies (IWPT). 2003: 195-206

[6].J.S. Ma, Y.Zhang, T.Liu, and S.Li. A statistical dependency parser of Chinese under small training data. Workshop:Beyond shallow analyses-Formalisms and statistical modeling for deep analyses, IJCNLP-04, San Ya. 2004:113-118

[7].Luong Nguyen Thi ,Dalat Univ,Lamdong,Vietnam ,Linh Ha My,Hung Nguyen Viet,Huyen Nguyen Thi Minh,Phuong Le Hong.Building a Treebank for Vietnamese Dependency Parsing[C]//IEEE RIVF International Conference on Computing and Communication Technologies - Research, Innovation, and Vision for the Future (RIVF), NOV 10-13, 2013

[8].Ryan McDonald. Non-projective dependency parsing using spanning tree algorithms. Association for Computational Linguistics. 2005: 523-530

[9].J. Nivre and M. Scholz. Deterministic Dependency Parsing of English Text. Proc. of the 20th Intern. Conf. on Computational Linguistics (COLING). 2004: 64-70

[10].Joakim Nivre, Ryan McDonald. Integrating GraphBased and transition-Based Dependency Parsers[C]. In Proc. of ACL, 2008:950-958