ICIC Express Letters
Part B: Applications
Volume **7**, Number **7**, July **2016**

ICIC International ©2016 ISSN 2185-2766
pp. **1645–1650**

BUILDING A TIBETAN SYNTACTIC AND SEMANTIC DEPENDENCY TREEBANK

Tashi Gyal[1] and Likun Qiu[2]

[1]Research Center of Tibetan Information Technology
Tibet University
No. 36, Jiangsu Road, Lhasa 850012, P. R. China
tashi77@163.com

[2]School of Chinese Language and Literature
Ludong University
No. 186, Middle Hongqi Road, Zhifu District, Yantai 264025, P. R. China
qiulikun@pku.edu.cn

Received December 2015; accepted March 2016

Abstract. *This paper takes dependency grammar as the theoretical basis to build a Tibetan syntactic and semantic dependency treebank. We first introduce the basic principles of dependency grammar. Then, we carefully analyze the selection of Tibetan sentences, formalized model of Tibetan dependency structure, and multidimensional aspects of Tibetan dependency treebank in detail. Further, we work out a syntactic and semantic annotation schema for Tibetan, and train a syntactic parsing system and a semantic parsing system on our newly-built Tibetan dependency treebank. Finally, we perform preliminary experiments, which demonstrate the effectiveness of the proposed annotation framework for Tibetan treebanking.*

Keywords: Dependency grammar, Tibetan treebank, Tibetan parsing

1. Introduction.

Dependency grammar mainly studies domination and subordination relations between each pair of words in a sentence. Dependency structure refers to the syntactic structure containing relations between words in a sentence, and can be represented in the form of tree structure, commonly called dependency tree. Knowledge of syntax and semantic structure is an important resource for natural language processing. In the meantime, it will be of great importance in the process of understanding natural language automatically. Due to its clear structure and simple form, both in syntactic and semantic levels, dependency grammar has attracted a lot of attention from scholars, and been widely used in building syntactic and semantic treebanks.

In natural language understanding, the annotation framework is very important in the process of building a dependency treebank. A well-formed framework is usually composed of a series of syntactic roles and semantic roles with a detailed instructions to demonstrate how to apply those roles to specific sentences. Zhou and Huang [1] reduced the number of syntactic relations for Chinese from 106 to 44 in order to lower the computing cost of automatic parsing. In contrast, we propose a novel annotation framework that contains 24 syntactic relations and 18 semantic relations based on the characteristics of Tibetan.

Compared with English and Chinese, natural language processing research on Tibetan is still at its preliminary stage. To our knowledge, there are no studies that have built Tibetan treebanks based on dependency grammar theory. However, this kind of resource is necessary for training high-quality syntactic parser for Tibetan, which has been proved to be useful in the processing of English, Chinese, etc. In this paper, we proposed an annotation framework for Tibetan syntactic and semantic dependency, and then built a Tibetan treebank to test the effectiveness of our framework. Preliminary experimental results show that our parser achieved a 79% UAS score trained on our small-scale treebank, which is very promising.

2. Theoretical Framework of Tibetan Dependency Treebank.

2.1. A short history of dependency grammar.

In academic community, the concept of dependency grammar can be traced back to the 4th century BC, founded by the Indian linguist Panini [2]. However, generally, Tesnière, a famous French linguist, is regarded as the founder of the theory of dependency grammar. In order to present a common grammatical theory, he conducted a comparative study in more than 10 languages, such as ancient Greek and Roman language. The core of dependency grammar theory is reflected in his famous book *Elements de Syntaxe Structurale*, and he first proposed the general theory of *syntactic structure* in this book. Later on, *syntactic structure* is referred to as *dependency grammar* or *affiliation grammar*. This theory explores typological similarities between various languages, and focuses on establishing cross-language system that is applicable to discover the deep syntax of human language objectively. This theory has made great influence on the development of general linguistics and computational linguistics.

2.2. Basic principles of dependency grammar.

Dependency grammar outlines relations between words in a sentence. The structure looks like a pyramid. A verb is taken as the structural center of the structure, and all other syntactic units are either directly or indirectly connected to the verb in terms of the directed links, which are called dependencies. Tesnière did not give a clear definition of dependency grammar, but he made the core essence of dependency theory through grammar verification process step by step. Based on his theory, many researchers presented their own understanding and interpretation upon dependency grammar.

3. Workflow of Tibetan Dependency Treebanking.

3.1. Principles of selecting Tibetan sentences.

When selecting sentences, we take the factors of type, genre, and era into consideration in order to ensure that the selected sentences would be representative and balanced. Traditional Tibetan grammatical theory divided Tibetan sentences into five types, namely transitive sentence, non-transitive sentence, dependency sentence, subject-predicate sentence and object-predicate sentence. The choice of sentences is based on the sentence type in order to cover various types of Tibetan sentences.

In terms of content, we choose a typical corpus that contains literature, academic articles, news, history, biography, religion and other main genres in Tibetan. Given the differences between Tibetan dialects, we only select sentences from written language. At the same time, contemporary characteristic is also regarded as an important factor in selecting sentences to ensure the representativeness. According to the above principles, we selected 10,000 sentences out of a large-scale Tibetan corpus to form a typical Tibetan corpus including single sentences, complex sentences and many other types of sentences.

3.2. The basic flow of treebanking.

First, we segment original Tibetan text into sentences and select sentences according to above principles. Second, automatic segmentation and POS tagging are performed to generate word-segmented and POS-tagged sentences. Third, each sentence is checked twice and revised if necessary. The proofreading process is conducted by four Tibetan graduate students to ensure the consistency of annotation. For word segmentation and POS tagging, we use *Segmentation Specification for Modern Tibetan Information Processing* and *POS Tag Set Specification for Modern Tibetan Language Information Processing*, which have been submitted to the Tibetan IT Standard Working Group of National Beacon Committee.

Second, we propose an annotation framework for building a dependency treebank based on segmentation and POS tagging. According to the Tibetan sentence classification, sentence structure and associated components were carefully labeled to form large-scale grammatical corpus with syntactic information to identify dependency relations so as to construct the dependency treebank.

4. **The Formalized Model of Tibetan Dependency Structure.**

Syntax is used to conduct analysis of language phenomena, through natural language to describe the characteristics of the objects involved, so as to achieve accurate representation of the theory. Another description of the object is to create a model, which is formed with an artificial structure by extracting certain characteristics from the object. In language, graph is widely used as a form of model, and the main elements of graph are vertices and edges, while *tree* is a special type of graph. In this way, we can clearly understand that language or the so-called formalized grammar is actually an approach to use symbol system to abstract the research objects. In other words, the understanding and generation of sentence are a linear sequence (one-dimensional) and a structure (two-dimensional) conversion process. In this process, the role of the schema cannot be ignored, because it can be expressed in an abstract, conceptual and vivid way [3]. In order to let computer mimic a one-dimensional linear to two-dimensional tree structure conversion process, models and formal methods can contribute to the study of language structure on computer. We can also say that formalization lays the foundation of a program.

4.1. **Tibetan dependency structure.**

The four axioms and five criteria are formal description of dependency grammar, and it is reasonable to use them to make formalization constrains for Tibetan dependency structure. In fact, Tibetan dependency relations refer to the domination and subordination relations between words, and the relations are of an unequal direction. The sentence structure is a top-down structure with hierarchy [4]. Generally, domination and subordination are described as a type of father-and-son relationship. In the process of Tibetan dependency parsing, syntactic dependency structure in the form of graph and symbol is taken as the bridge to connect dependency grammar and parsing algorism. It will be formalized in the form of grammar rules or constraints to describe various information nodes attached to the side. Common Tibetan dependency structure schema includes directed graph, Tibetan dependency tree, case-marked tree, and Tibetan projective dependency tree.

In Figure 1, different labels are used to represent different parts of speech. For instance, *nr* denotes person name, *bo* represents an object case, *nn* represents a general term, *ba* represents a tool case, *ls* represents an industry case, and *vt* represents a transitive verb. Relations between two words are represented by an arc between them with an arrow pointed to the subordinate word, which makes it possible for us to better understand the dependency between the words in hierarchy.

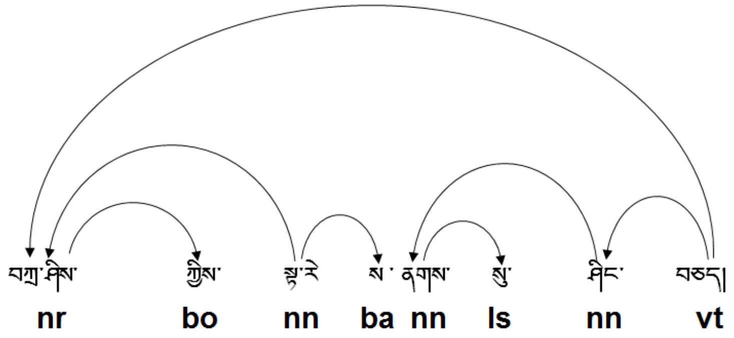

Figure 1. A directed graph

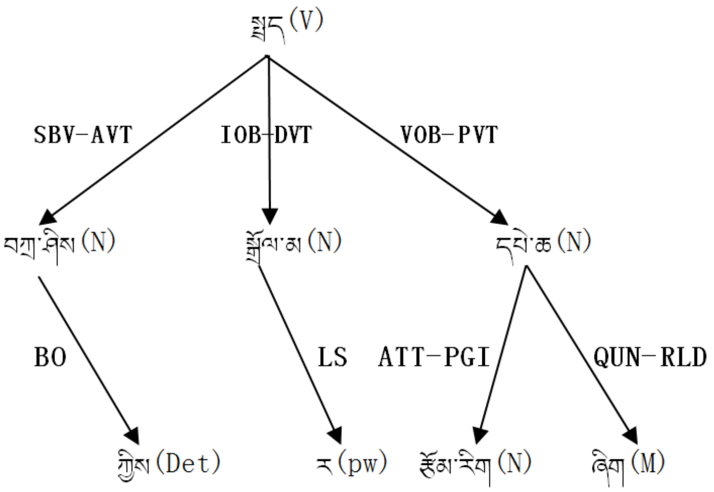

FIGURE 2. Dependency tree

According to the results of dependency analysis, a three-dimensional tree structure with multiple labels is shown in Figure 2.

The figure above should be interpreted in syntactic and semantic levels, respectively. In the first layer, "སྤྲད" is a verb. In the second layer "བཀྲ་ ཤིས" is the first action element, served as subject (SBV) and agent (ATV). "སློབ་ མ" is the second action element, served as a first object (IOB) and objects (DVT). "དཔེ་ ཆ" is the third action element, served as the second object (VOB) and objects (PVT) in semantic sense. In the third layer, there are three status elements of subordinate action elements, with their disposable unit formed an integral agent, object and predicate. "ཀྱིས" is a tool case marker (BO), following a subject or agent. "ར" is an adhesion mark (LS), which bounds to the first object and belongs to object semantically, and the makers of these two cases are at the right of a noun. "རྩོམ་ རིག" is a noun, belongs to the second object or agent "དཔེ་ ཆ". "ཞིག" is a quantifier, which is behind the second object, and forms an integral phrase served as object semantically.

4.2. **Multidimensional analysis of Tibetan dependency tree.** In a dependency tree, there are four types of information: words, POS tags, syntactic dependencies, and semantic dependencies, as shown in Figure 3 and Figure 4.

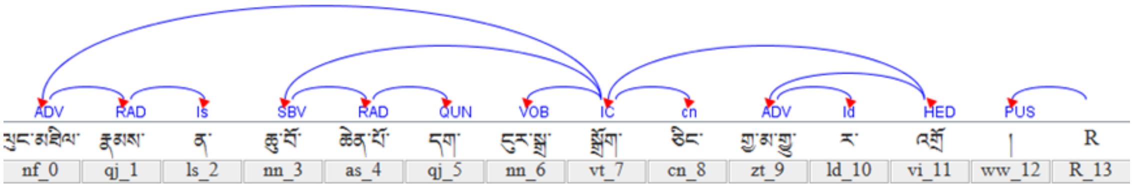

FIGURE 3. Tibetan syntactic dependency tree

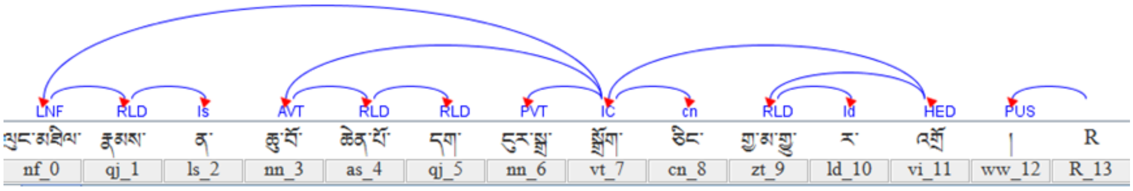

FIGURE 4. Tibetan semantic dependency tree

A syntactic dependency tree is shown in Figure 3. The first layer indicates POS tags and their ID in the current sentence, the second layer is a list of words, and the third layer consists of a set of arcs, each of which points from the head word to its dependent word and is annotated with a syntactic relation tag.

A semantic dependency tree is shown in Figure 4. It is quite similar to the sentence in Figure 3. They differ in that the syntactic relation tags in Figure 4 have been replaced with semantic relation tags.

5. Experiments.

5.1. Experimental setup.

Data. We constructed a Tibet dependency treebank (TDT) that contains 1200 sentences. In the following experiments, the previous 1100 sentences are used as training set, and the last 100 sentences are used as testing set, respectively. In the training process, all the default parameters are used, and thus we don't need a development set.

Dependency Parser. To show the effectiveness of the proposed treebank, we train a Tibetan syntactic dependency parser and then test it using MATE-Tools 3.6.1, which can be downloaded from https://code.google.com/p/mate-tools/. MATE-Tools [5] is one of the state-of-the-art dependency parsers and can support multi-thread training. It is comparable in accuracy to another famous parser, i.e., ZPar [6,7], and remarkably better than the other two parsers, i.e., MSTParser and MALTParser [8]. For the semantic dependency parsing, we use the semantic role labeling module of MATE-Tools 3.6.1 [9]. This semantic role labeler achieved the best results on the Chinese data of CoNLL2009 Shared Task on Semantic Role Labeling.

Evaluation Metrics. The accuracies of dependency parsing are calculated using the evaluation metrics of the CoNLL 2009 shared task scorer [10], which evaluates the accuracy of syntactic dependency parsing with UAS (unlabeled attachment score) and LAS (labeled attachment score), and evaluates the accuracy of semantic dependency parsing with labeled precision, recall and F1.

5.2. Experimental results.

The MATE-Tools syntactic dependency parser achieved 79.18% UAS and 70.82% LAS, and the semantic parser achieved 75.85% F1 over automatic syntactic parsing. This scores are very promising, although the number of sentences in our treebank is relatively small. If the scale of treebank becomes larger, both the two accuracies may be improved accordingly.

6. Conclusion.

Tibetan treebanking is a key step in Tibetan natural language processing, and performs important intermediate functions between Tibetan syntactic and semantic analysis. This paper takes dependency grammar as the theoretical framework to build a Tibetan treebank, aiming to promote the development of Tibetan information processing. Our experiments show that the newly-built Tibetan dependency treebank achieves good results in statistic parsing, although the accuracy is expected to be further improved with a much larger Tibetan treebank.

Acknowledgments. We gratefully acknowledge the anonymous reviewers for their constructive comments, and the support of the National Natural Science Foundation of China (Nos. 61163043, 61572245), Major National Social Science Fund of China (No. 14ZDB101), Innovative Support Program for Young Teachers of Tibetan Universities (QC2015-19), and Zhufeng Scholar Project.

REFERENCES

[1] M. Zhou and C. Huang, A Chinese dependency system for corpus annotation, *Journal of Chinese Information Processing*, vol.8, no.3, pp.35-50, 1994.

[2] C. Zong, *Statistical Natural Language Processing*, Tsinghua University Press, 2008.

[3] H. Liu, *Theory and Practice of Dependency Grammar*, Science Press, 2009.

[4] Y. Ming, *Editor Classics of Western Linguistics Readings*, China Renmin University Press, 2011.

[5] B. Bohnet, Top accuracy and fast dependency parsing is not a contradiction, *Proc. of COLING*, pp.89-97, 2010.

[6] L. Qiu, Y. Zhang, P. Jin and H. Wang, Multi-view Chinese treebanking, *Proc. of COLING*, pp.257-268, 2014.

[7] Y. Zhang and S. Clark, Syntactic processing using the generalized perceptron and beam search, *Computational Linguistics*, vol.37, no.1, pp.105-151, 2011.

[8] W. Che, V. Spitkovsky and T. Liu, A comparison of Chinese parsers for Stanford dependencies, *Proc. of EACL*, pp.11-16, 2012.

[9] A. Björkelund, L. Hafdell et al., Multilingual semantic role labeling, *Proc. of CoNLL*, pp.43-48, 2009.

[10] J. Hajic, M. Ciaramita et al., The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages, *Proc. of CoNLL*, pp.1-18, 2009.