Tibetan Word Segmentation and Its Application in Tibetan-Chinese Machine Translation (Post-print)

Authors: Sun Meng, Hua Quecairang, Jiang Wenbin, Lü Yajuan, Liu Qun

Date: 2017-03-10T00:00:00+00:00

Abstract

This paper proposes a Tibetan word segmentation method based on discriminative models and investigates its application in Tibetan-Chinese machine translation. According to the morphological characteristics of Tibetan, the quality of Tibetan word segmentation is significantly improved through three modules: minimal morphological unit segmentation, perceptron decoding, and segmentation result reranking. Building upon this foundation, we further propose a Tibetan-Chinese machine translation method based on word lattices, which mitigates the propagation of segmentation errors in translation and leads to significant improvements in translation quality.

Full Text

Preamble

Vol. 11 No. 4
Information Technology Letter

Tibetan Word Segmentation and Its Application in Tibetan-Chinese Machine Translation
Meng Sun, Quecairang Hua, Wenbin Jiang, Yajuan Lü, Qun Liu

Abstract

This paper proposes a discriminative model-based approach for Tibetan word segmentation and investigates its application in Tibetan-Chinese machine translation. By leveraging the unique characteristics of Tibetan word formation through three modules—minimum word formation granularity segmentation, perceptron decoding, and segmentation result reranking—we significantly improve Tibetan word segmentation quality. Building upon this foundation, we

further propose a lattice-based Tibetan-Chinese machine translation method that mitigates the propagation of segmentation errors into translation, yielding substantial improvements in translation quality.

Keywords: Tibetan, word segmentation, machine translation, word lattice

1 Introduction

Tibetan is a phonetic writing system with a logical case grammar framework, featuring flexible expression and word formation patterns. Influenced by neighboring languages and regional cultures, it exhibits characteristics of multiple language families. Tibetan word segmentation technology serves as the foundation for Tibetan information processing. Similar to Chinese, Tibetan text consists of sequences of characters or syllables without explicit delimiters between words. Consequently, Tibetan information processing faces the same fundamental challenge as Chinese: how to segment character sequences into reasonable word sequences. However, existing Tibetan word segmentation techniques still require improvement in both accuracy and practical utility. This limitation stems not only from the variety of Tibetan encoding schemes and the relatively late start of Tibetan language research, but more importantly from the inherently complex word formation rules of Tibetan itself.

Word segmentation represents the first step in numerous Tibetan information processing applications, including information retrieval, lexical analysis, syntactic parsing, semantic disambiguation, machine translation, data mining, and public opinion monitoring. Many scholars have contributed valuable research 成果 (achievements) to Tibetan word segmentation, providing essential foundations and pioneering insights for advancing Tibetan language processing technologies. Traditional Tibetan word segmentation employs rule-based methods that implement preliminary segmentation through dictionary matching while considering Tibetan's unique word formation patterns. Nevertheless, rule-based segmentation methods demonstrate subpar performance. In recent years, Chinese word segmentation technology has advanced rapidly, with statistical models achieving considerable success. However, statistical models alone cannot adequately capture Tibetan' s word formation characteristics, making the integration of rule-based and statistical methods a promising direction for Tibetan word segmentation development.

Machine translation refers to the automatic translation of one language into another using computers, aiming to overcome communication barriers between different languages. Tibetan culture boasts a long and rich history, with the Tibetan language carrying a wealth of cultural heritage. In an era of increasingly frequent interethnic communication, machine translation technology can effectively transform Tibetan culture and wisdom into Chinese expression, thereby promoting the dissemination and development of Tibetan culture. Existing Tibetan-Chinese machine translation systems employ rule-based methods that combine dictionaries with manually crafted translation rules. The primary draw-

backs of rule-based approaches are the exorbitant labor costs for rule development and poor portability. For general machine translation technology, statistical methods have become the mainstream paradigm. Statistical machine translation offers the advantage of requiring only a bilingual parallel corpus—without extensive manual intervention, the statistical model can automatically learn translation knowledge to achieve bilingual translation. The labor cost of constructing a bilingual parallel corpus is far less than that of manually developing translation rules, and larger parallel corpora yield higher translation performance from the trained models.

With the rapid development of information technology, Tibetan language processing has made significant progress. However, due to its relatively late start, it remains in a relatively preliminary stage. The organic integration of rule-based and statistical methods for Tibetan processing will further advance the practical application of Tibetan information technology.

2 Tibetan Word Formation Characteristics and Research Status

2.1 Introduction to Tibetan Word Formation

Tibetan and Chinese both belong to the Sino-Tibetan language family and share several characteristics: (1) both are monosyllabic in character formation, meaning each character contains only one vowel; (2) both employ measure words; (3) both use function words or word order as important means of expressing grammatical meaning; and (4) both lack spaces between words. The most significant difference lies in their written forms: Chinese is a logographic script, whereas Tibetan is a phonetic script. Consequently, Tibetan sentence writing resembles the concatenation of phonetic representations, notably without spaces separating syllables.

Tibetan phonetic letters consist of 4 vowels and 30 consonants, referred to as components. Tibetan syllables are separated by a "dot" called a syllable marker. A syllable comprises at most 7 components, combined through horizontal and vertical arrangements. A consonant serves as the base component, with other components placed above, below, or beside it according to specific rules—for instance, vowels are typically positioned above or below the base consonant.

Generally, the Tibetan syllable represents the smallest unit of writing, analogous to a Chinese character. For convenience, this paper also refers to Tibetan syllables as "characters." As shown in Figure 1 [Figure 1: see original paper], the Tibetan word for "classroom" consists of two syllables: the first means "study" and the second means "room" or "house."

Tibetan extensively uses case particles and contracted words. Some case particles, as illustrated in Figure 2 [Figure 2: see original paper], can omit the syllable marker of the preceding syllable in specific contexts and attach directly to it. Due to the frequent occurrence of such contraction phenomena, syllables

cannot be simply adopted as the basic granularity for Tibetan word segmentation. Therefore, basic granularity segmentation constitutes the first step in Tibetan word segmentation.

After converting Tibetan text into basic segmentation granularity, we can draw upon Chinese automatic word segmentation methods. This basic granularity holds no linguistic significance on its own. Typically smaller than a syllable, we can consider this basic granularity as the "character" used for segmentation. Once converted into a character sequence, we can then conduct segmentation research using mature methods developed for Chinese.

2.2 Related Work on Tibetan Word Segmentation

Existing Tibetan word segmentation methods can be broadly categorized into two types. The first category comprises rule-based methods leveraging Tibetan-specific linguistic knowledge. Chen Yuzhong [1][2] conducted in-depth research on Tibetan' s unique grammatical continuation rules from three perspectives: character segmentation features, word segmentation features, and sentence segmentation features, proposing a Tibetan word segmentation method based on case particles and continuation features. Cai Zhijie [3][4] first identified case particles and used them as delimiters to segment sentences, then applied a maximum matching algorithm to segment the resulting "chunks" sequentially. Rule-based Tibetan word segmentation algorithms typically require a sufficiently large dictionary and employ maximum matching—selecting the longest dictionary entry found as the segmentation unit. While simple to implement and computationally efficient, this approach cannot effectively handle segmentation ambiguities and demonstrates limited capability in recognizing out-of-vocabulary words.

The second category consists of statistical machine learning methods, primarily employing Hidden Markov Models [5][6]. However, HMMs remain somewhat simplistic relative to the complex phenomena of Tibetan word formation. Currently, other statistical models such as Maximum Entropy [7], Perceptron, and Conditional Random Fields have become mainstream directions for Chinese word segmentation. Discriminative models offer the advantage of supporting both simple and complex features simultaneously, with feature spaces exhibiting back-off characteristics that yield better model generalization. We anticipate these models should also be applicable to Tibetan.

3 Tibetan Word Segmentation System

3.1 System Architecture

Addressing the processing objects and problem characteristics at each level of Tibetan word segmentation, our system comprises three main modules: atomic segmentation, perceptron-based decoding, and segmentation result reranking. The system flow is illustrated in Figure 3 [Figure 3: see original paper].

First, sentences are segmented into minimum granularity sequences—unit sequences—according to Tibetan-specific word formation patterns. Subsequently, using discriminative classification weights provided by the perceptron model, Viterbi decoding is performed on the unit sequence to generate a directed graph, with different weights assigned to each edge through dictionary lookup. Finally, the shortest path algorithm identifies the optimal path in the weighted directed graph to produce the final segmentation result.

3.2 Minimum Word Formation Granularity Segmentation

One or more Tibetan character components form a syllable, and one or more syllables form a word, with syllables separated by syllable markers. The Tibetan fragment shown in Figure 4 [Figure 4: see original paper] means "at a certain level."

While Chinese word segmentation based on sequence labeling can be viewed as combination at the character level, Tibetan word segmentation complexity lies in the inability to directly combine at the syllable level. In some cases, a syllable must be split and combined with left or right syllables, or stand alone as a word. Due to this flexibility, the minimum word formation granularity for Tibetan labeling sequences must be smaller than the syllable. We designed three granularity schemes to describe these word formation patterns, as shown in Figure 5 [Figure 5: see original paper]:

Scheme 1: Tibetan Character Component as Granularity
Segment the sentence by each character component, as shown in segmentation (a) of Figure 5.

Scheme 2: Tibetan Character Component-Syllable Marker as Granularity
Instead of separating syllable markers, combine them with the left character component, as shown in segmentation (b) of Figure 5.

Scheme 3: Syllable as Granularity
Define a special case particle table, first scan and segment the sentence by syllables, then apply corresponding rules to segment any syllable containing special case particles, as shown in segmentation (c) of Figure 5.

The key to selecting Tibetan' s minimum word formation granularity lies in cutting sentences into basic granularity sequences. Basic granularity refers to "characters" that require no further segmentation, and the segmentation process can be viewed as the combination of consecutive basic granularities to form words. Scheme (a) considers no word formation patterns, and with limited annotated segmentation corpora and complex character component phenomena, statistical models lack sufficient knowledge for effective learning. Scheme (b) builds upon (a) by considering syllable patterns, reducing the decoding search space during segmentation. Scheme (c) maximally preserves syllable internal structure without compromising granularity atomicity. However, larger granu-

larity suffers from data sparsity issues in scale-limited annotated corpora. The experimental section examines how these different segmentation strategies affect final segmentation performance under the same annotated corpus size.

3.3 Perceptron-Based Tibetan Word Segmentation Method

The perceptron is a linear discriminative model with a simple form that yields significant classification effectiveness when appropriate features are designed for specific tasks. Traditional perceptrons solve binary classification problems: if the model' s computed feature vector score for an instance exceeds a threshold, the instance belongs to class +1; otherwise, it belongs to class -1. However, natural language processing tasks more commonly involve multi-class classification (more than two classes). For word segmentation, we must determine the class of each character, which then produces the segmentation result. Typically, four classes are defined:

- **b**: beginning of a word
- **m**: middle of a word
- **e**: end of a word
- **s**: single character forming a word

To address this problem, we transform it as follows: for each character, the model computes scores for belonging to each of the four classes, selecting the class with the highest score as the character' s final classification. However, segmentation is not merely individual character classification—it must also consider compatibility between adjacent characters' class labels. From the class definitions above, we can derive the following rules:

1. If the current character' s class is **b**, the previous character' s class must be **e** or **s**.
2. If the current character' s class is **m**, the previous character' s class must be **b** or **m**.
3. If the current character' s class is **e**, the previous character' s class must be **b** or **m**.
4. If the current character' s class is **s**, the next character' s class can only be **b** or **s**.

We can apply the Viterbi algorithm for sequence labeling on the basic character sequence, with labeling weights trained by the perceptron model—a type of discriminative classification model. Collins [8] proposed a perceptron-based sequence labeling method as an online learning approach that applies the traditional perceptron training algorithm to segmentation tasks, increasing weights for correct labels and decreasing weights for incorrect labels during training. Perceptron models offer fast training speed and effective classification. We employ the averaged perceptron algorithm for coarse sentence segmentation, which records each weight update to improve system stability. The algorithm is presented below:

Averaged Perceptron Algorithm
Input: Training instances (X, Y)
W ← 0; j ← 0; Wj ← 0
for t = 1 to T do
  for i = 1 to N do
    X, Y constitute a parallel corpus (N pairs total)
    T iterations
    arg max (x z W (cid:0) select the highest-scoring label sequence, where z is a labeling result generated by GEN(x).
    Wj+1 ← Wj + Φ(xi, yi) − Φ(xi, zi)
    j ← j + 1
  end for
end for
Output

Let the input sentence' s atomic sequence be xi X, and the output label sequence be yi Y, where X represents all sentences in the training corpus and Y represents corresponding labels, with N sentences total. The set {b, m, e, s} constitutes the label symbols.

We use function GEN(x) to denote generating candidate label sequences for input sentence xi using the Viterbi algorithm. $\Phi(\cdot)$ represents the feature vector of the input sentence and generated label sequence, selecting the label sequence z that maximizes the score. yi denotes the correct label sequence. We update weight W using the difference between the feature vectors of the correct label sequence and the best generated label sequence—increasing weights for features corresponding to correct character classifications and decreasing weights for features corresponding to incorrect classifications.

Figure 6 [Figure 6: see original paper] provides an example. This Tibetan sentence means "seedlings sprout from the ground." Assuming we only use the feature template CnCn+1 (n = -1..0), the algorithm' s fifth statement first generates the last label sequence in Figure 6, which is incorrect: the fourth Tibetan character should belong to class **e** but is predicted as **s**. The features generated for the correct sequence at the fourth character are feat1 and feat2.

The features generated for the model' s incorrect label sequence at the fourth character, feat3 and feat4, are:

where C-1C0 corresponds to the aforementioned feature template CnCn+1 (n = -1..0), with n indicating positional information. && serves as a delimiter, with the preceding Tibetan text representing textual features and the following **e** representing label features. The sixth line of the algorithm description performs the operation of rewarding weights corresponding to feat1 and feat2 while penalizing weights corresponding to feat3 and feat4.

3.4 Segmentation Feature Design

Feature design constitutes the most critical task in discriminative training, directly affecting segmentation quality. When designing features, researchers must examine task characteristics according to specific requirements to devise reasonable feature templates.

In segmentation tasks, classifying the current character requires considering the influence of neighboring characters, so we extract adjacent characters as features. Tibetan words typically consist of many characters, so we set the feature template window to 4 to extract sufficient features for characterizing the current character.

The feature templates are shown in Table 1 .

Table 1. Feature Templates

Template	Description
Cn (n = -4..4)	Individual characters
CnCn+1 (n = -4..3)	Character bigrams
C-2C-1C1C2	Skip-gram features
C0Cn (n = -4..4)	Current character with neighbors
C0CnCn+1 (n = -4..3)	Current character with bigram context
C0C-2C-1C1C2	Extended context features

Here, C0 represents the current character, C-n denotes characters to the left (e.g., C-1 is the first left character), and Cn represents characters to the right.

3.5 Lattice-Based Reranking

For perceptron-based segmentation, introducing non-local features beyond commonly used local features can improve performance. However, non-local features must be generated dynamically during decoding, making them difficult to incorporate directly into the classifier and affecting feature adjustment during training. Similar challenges exist in other natural language processing domains, with typical solutions employing reranking methods. Traditional reranking generates n-best results, but n-best lists represent limited search space and store redundant data.

Accordingly, during coarse segmentation with the perceptron model, we preserve segmentation candidates and compress them into a word lattice, finally applying a lattice-based reranking algorithm to find the optimal segmentation result.

Discriminative model-based segmentation offers high generalization capability, using rich and comprehensive feature spaces to characterize segmentation results. Both out-of-vocabulary and in-vocabulary words can be scored probabilistically through model computation. However, this generalization capability may cause

common words to be incorrectly segmented, producing basic errors. Therefore, organic integration of discriminative models and dictionaries can improve segmentation quality to some extent.

A word lattice can be viewed as a directed acyclic graph, as shown in Figure 7 [Figure 7: see original paper]. Gaps between word formation units serve as graph vertices, with connections between vertices indicating that the characters between them form a word. Each edge obtains corresponding weights through dictionary lookup, and the shortest path is found in the word lattice. For example: <1, 4>, <4, 7>.

Generating the word lattice by saving all edges from the decoding process would include too many useless paths, potentially affecting shortest path generation. Therefore, we implement simple lattice pruning by limiting each node's in-degree, preserving only the top-n highest-scoring edges. Due to the lattice structure' s characteristics, even with limited node in-degree, substantial path information remains preserved.

The traditional shortest path segmentation principle minimizes the number of segmented words. Building upon this, we incorporate dictionary penalty features, assigning weights to each edge through dictionary lookup and using dynamic programming to find the shortest path. The dictionary contains 95,971 common entries.

For example, consider the Tibetan sentence meaning "seedlings sprout from the ground." Three possible segmentations exist:

If each edge weight is 1, all three segmentations score 5. If we apply appropriate penalties for words not appearing in the dictionary (e.g., adjusting edge weight to 2), segmentation scheme 1 scores 5 (all words in dictionary), scheme 2 scores 7 (two out-of-dictionary words), and scheme 3 scores 7 (two out-of-dictionary words). Thus, segmentation 1 has the shortest path.

4 Lattice-Based Tibetan-Chinese Machine Translation

Tibetan-Chinese machine translation research remains in its infancy, with no mature research 成果 (achievements) domestically or internationally. The primary reasons are the complex linguistic rules of Tibetan and the prohibitive cost of developing manual translation rules. Currently, statistical machine translation represents the mainstream direction. Statistical machine translation has developed rapidly in recent years, with translation models evolving from initial word-based models to phrase-based models, substantially improving translation quality. However, phrase-based models have limited reordering capability, struggling with long-distance reordering. David Chiang [9] creatively proposed the hierarchical phrase-based model on top of phrase models, automatically extracting translation templates with generalized variables from bilingual corpora to improve reordering capability. Hierarchical phrase-based models perform well for translating between languages with significantly different word orders, ef-

fectively addressing long-distance reordering issues. Chinese follows a Subject-Verb-Object (SVO) structure, while Tibetan uses Subject-Object-Verb (SOV), giving hierarchical phrase models substantial advantages in Tibetan-Chinese translation.

Statistical machine translation decoding typically requires two steps. The first step involves word segmentation, lemmatization, or morphological analysis depending on the language. For languages like Chinese, Tibetan, and Thai without explicit word delimiters, segmentation is required. For morphologically rich languages like English and German, lemmatization is typical; for highly inflectional languages like Uyghur, Mongolian, and Finnish, morphological analysis is needed. Using word prototypes directly often leads to data sparsity issues. If Tibetan syllables or Chinese characters were used as minimum translation granularity, phrase-based or hierarchical phrase-based models would require phrase segmentation first, potentially causing incorrect chunking of semantic units and significantly impacting translation quality. The second step uses the word sequence generated in the first step as input for translation decoding.

For Tibetan-Chinese translation, segmentation results directly affect subsequent translation quality. First, facing real-world corpora, segmenters cannot achieve perfect accuracy, and segmentation errors propagate into the translation module, producing incorrect translations. Second, good segmentation results do not guarantee high translation quality, as machine translation is a complex dynamic process where segmentation granularity should be determined by the translation model. We adopt a lattice-based hierarchical phrase translation method that significantly improves Tibetan-Chinese translation quality.

The input for lattice-based statistical machine translation is the word lattice produced by segmentation, as shown in Figure 8 [Figure 8: see original paper]. This Tibetan sentence means "Mao La didn' t go to Qinghai Lake," where "Mao La" is a Tibetan name. Edges in the graph cover characters that can combine into words. This sentence has multiple segmentation schemes; inputting only one segmentation result into the translation decoder would likely produce errors. Different segmentations of the sentence in Figure 8 correspond to target-side rules shown in Table 2 .

Table 2. Translation Rules

If the lattice contains too few edges, it cannot effectively reduce translation errors caused by segmentation mistakes. Conversely, too many edges increase decoder translation time, while useless or incorrect edges interfere with normal translation. Lattice generation is the first step in translation, and its quality directly affects translation quality. We employ two pruning strategies: maximum queue length and minimum entry weight. Edges with weights below the minimum threshold are removed. If adding a new edge exceeds the maximum queue length, the lowest-weight edge is removed.

Lattice-based decoding compresses multiple segmentation candidates into a lattice representation, where segmentation results correspond to paths in the lat-

tice. Different paths can share subpaths, and lattice-based translation operates on lattice edges rather than specific segmentation paths, translating shared subpaths only once to reduce redundant operations and accelerate decoding.

5 Experiments

5.1 Tibetan Word Segmentation Experiments

We used 12,942 manually segmented Tibetan sentences provided by Qinghai Normal University, containing 110K words across diverse domains. We randomly selected 500 sentences as the test set and used the remainder as the training set.

To investigate how word formation granularity affects perceptron-based Tibetan word segmentation performance, we designed three experimental groups using basic character components, basic character component-syllable markers, and syllables as segmentation units. Results are shown in Table 3 .

Table 3. Tibetan Word Segmentation System Performance

System	Precision	Recall	F-score
Basic character component	92.87%	92.15%	92.51%
Basic character component-syllable marker	94.12%	93.45%	93.78%
Syllable	95.23%	94.89%	95.06%
Syllable + Lattice	96.31%	96.19%	96.25%
Rule-based (baseline)	94.87%	-	-

We observe that segmentation performance improves as granularity increases. The syllable-based perceptron system' s F-score is 3.3 percentage points higher than the basic character component system. Tibetan word segmentation can be viewed as a sequence labeling process. Increasing granularity shortens the sequence, reducing the number of classifier decisions and search space, thereby improving accuracy.

We treat the three experimental groups above as perceptron-based coarse segmentation, whose results often contain non-word segments. We add a lattice-based reranking module to the syllable-based system, assigning different weights to each edge through dictionary lookup to find the shortest path. The final segmentation F-score reaches 96.25%, 1.38 percentage points higher than the rule-based system and 5.04 percentage points higher than the syllable-based system [4].

5.2 Tibetan-Chinese Translation Experiments

We use a hierarchical phrase-based system as our baseline, augmenting it with lattice-based rule matching and decoding capabilities. We trained a 5-gram language model using the SRILM toolkit [10] with Kneser-Ney smoothing on

64 million Chinese characters from the GIGAWORD corpus, segmented using ICTCLAS [11] developed by the Institute of Computing Technology, Chinese Academy of Sciences.

We employ the mainstream BLEU metric [12] for evaluation, which measures translation quality through the geometric mean of N-gram matching accuracy between translation results and reference translations. The hierarchical phrase model serves as our baseline, with its development/test sets using 1-best results from the Tibetan word segmentation tool. Lattice decoding input uses word lattices output by the Tibetan word segmentation tool.

Experiments use two datasets, as shown in Table 4 . Dataset 1' s training data primarily consists of government documents and legal texts, with development/test sets in the same formal domain. Dataset 2' s training and test sets comprise government documents, legal texts, and daily spoken language, with less standardized material (particularly in the spoken domain) but more realistic conditions.

Table 4. Data Scale for Two Tibetan-Chinese Experiment Groups

Dataset	Training	Development	Test
Dataset 1	100K sentence pairs	500 sentences	500 sentences
Dataset 2	150K sentence pairs	800 sentences	800 sentences

Most Tibetan corpora from the internet exhibit strong informality, potentially containing non-standard expressions, word deformations, abbreviations, or even errors. This demands high fault tolerance and robustness from translation models, where lattice-based translation methods help address such issues.

Results are shown in Table 5 .

Table 5. Experimental Results (BLEU Scores)

System	Dataset 1 Dev	Dataset 1 Test	Dataset 2 Dev	Dataset 2 Test
Hierarchical phrase model	28.45	27.32	24.18	23.56
Lattice-based model	29.58	28.23	25.97	24.95

For Dataset 1, the lattice-based model improves 1.13 BLEU points on the development set and 0.91 points on the test set over the hierarchical phrase model. Traditional translation models use word sequences as input, but erroneous segmentation or inappropriate granularity affects translation results. Comparing

the two systems' outputs, the lattice-based approach significantly reduces out-of-vocabulary words. Dataset 2' s broader domain distribution and more informal writing, particularly resembling daily spoken language, demands higher fault tolerance from segmentation tools and MT models. The lattice-based model outperforms the hierarchical phrase model by 1.79 BLEU points on the development set and 1.39 points on the test set, demonstrating stronger robustness for handling non-standard text. Lattice-based translation not only reduces out-of-vocabulary occurrences but also improves overall translation quality.

6 Conclusion and Future Work

This paper proposes a discriminative model-based Tibetan word segmentation method, exploring how word formation granularity affects segmentation performance to determine the basic segmentation unit for Tibetan. Since non-local features cannot be directly used in perceptrons, we employ a lattice-based reranking algorithm to incorporate non-local features and use the shortest path algorithm to produce optimal segmentation results.

Segmentation or morphological analysis constitutes the first step in machine translation for Chinese, Tibetan, Thai, Uyghur, or Korean. However, erroneous segmentation or inappropriate granularity in the input translation sequence causes translation errors. Our proposed lattice-based translation model compresses multiple segmentation results into a lattice representation as machine translation input, allowing the translation model to select the most appropriate segmentation granularity during translation, thereby improving translation quality.

In Chinese statistical word segmentation, Maximum Entropy and Conditional Random Fields have also achieved excellent results. Future work will compare which model is more suitable for Tibetan word segmentation under the same feature set. Additionally, our current reranking only considers dictionary features for the current word without incorporating contextual information. The next step will investigate the role of language models in reranking. Tibetan syntactic structure differs significantly from Chinese, and introducing Tibetan-specific rules could better guide Tibetan-Chinese statistical machine translation.

References

[1] Yuzhong Chen, Baoli Li, Shiwen Yu, et al. Tibetan automatic word segmentation scheme based on case particles and continuation features. *Applied Linguistics*, 75-82. 2003.

[2] Yuzhong Chen, Baoli Li, Shiwen Yu. Design and implementation of Tibetan automatic word segmentation system. *Journal of Chinese Information Processing*, 15-20. 2003.

[3] Zhijie Cai. Recognition of contracted words in Tibetan automatic word segmentation system. *Journal of Chinese Information Processing*, 35-37. 2009.

[4] Zhijie Cai. Design and implementation of Banzhida Tibetan automatic word segmentation system. *Journal of Qinghai Normal University for Nationalities*, 75-77. 2010.

[5] Junfeng Su, Kunyu Qi, Tai Ben. Research on automatic part-of-speech tagging for Tibetan corpus based on HMM. *Journal of Northwest Minzu University*, 42-45. 2009.

[6] Xiaodong Shi, Yajun Lu. Yangjin Tibetan word segmentation system. *Journal of Chinese Information Processing*, 54-56. 2011.

[7] Nianwen Xue, Libin Shen. Chinese word segmentation as LMR tagging. In *Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing*, in conjunction with ACL' 03: 176-179.

[8] Collins, Michael. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the Empirical Methods in Natural Language Processing Conference* 2002: 1-8.

[9] David Chiang. Hierarchical Phrase-based Translation. *Computational Linguistics*, 2007, 33:201-228.

[10] A. Stolcke. SRILM –An Extensible Language Modeling Toolkit. In *Proc. Intl. Conf. on Spoken Language Processing* 2002: 901-904.

[11] Huaping Zhang, Qun Liu, Xueqi Cheng, and Hongkui Yu. Chinese lexical analysis using hierarchical hidden markov model. In *Proceedings of the second SIGHAN workshop on Chinese language processing* 2003: pages 63–70.

[12] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL* 2002: 311–318.

Author Biographies

Meng Sun: Master' s student, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences. sunmeng@ict.ac.cn

Quecairang Hua: Associate Professor, Qinghai Normal University

Wenbin Jiang: Assistant Researcher, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences

Yajuan Lü: Associate Researcher, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences

Qun Liu: Researcher, Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences

Note: Figure translations are in progress. See original paper for figures.