JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 30, 1407-1424 (2014)

Woodpecker: An Automatic Methodology for Machine Translation Diagnosis with Rich Linguistic Knowledge*

BO WANG[1], MING ZHOU[2], SHUJIE LIU[2], MU LI[2]
AND DONGDONG ZHANG[2]
[1]*School of Computer Science and Technology*
Tianjin University
Tianjin, 300000 P.R. China
[2]*Microsoft Research Asia*
Beijing, 100000 P.R. China
E-mail: bo.wang.1979@gmail.com; {*mingzhou*; *shujieli*; *muli*; *dozhang*}*@microsoft.com*

Different from the "black-box" evaluation, the diagnostic evaluation aims to provide a better explanatory power into various aspects of the performance of artificial intelligence systems. However, for machine translation (MT) systems, due to its complexity and knowledge dependency, such diagnostic evaluation often demands a large amount of manual work. To tackle this problem, we propose an automatic diagnostic evaluation methodology, called Woodpecker, which enables multi-factored evaluation of MT systems based on linguistic categories and automatically constructed linguistic checkpoints. The taxonomy of the categories is defined with rich linguistic knowledge, including phenomena on different linguistic levels. The instances of the categories are composed into test cases called linguistic checkpoints. We present a method that automatically extracts checkpoints from parallel sentences, through which, Woodpecker can automatically monitor a MT system in translating various linguistic phenomena, thereby facilitating diagnostic evaluation. The effectiveness of Woodpecker is verified through in-house experiments and open MT evaluation tracks on various types of MT systems.

Keywords: evaluation, diagnosis, machine translation, linguistic knowledge, checkpoint

1. INTRODUCTION

Evaluation is very important for artificial intelligence system development. The most popular evaluation is "black-box" style described in Turing test. In Machine Translation (MT) study, a great deal of effort has been devoted to automatic general evaluation metrics, *e.g.*, BLEU [1], Rouge [2] and METEOR [3]. Nevertheless, the development of MT also demands a good insight into the alternations of the system performance. However, this kind of diagnostic evaluation has not been well addressed: though the present metrics can break the overall score down into several finer-grained scores, the numbers of the scores are very small and are not very well explained. A **diagnostic Evaluation** is an evaluation metric which can diagnose the MT systems by providing feedback on the ability of MT systems to translate various aspects of the language, and thus reveal the strength and weakness of a MT system, benefitting the targeted development.

Many automatic evaluation metrics have been proposed to MT systems. Metrics

Received June 11, 2013; revised July 31, 2013; accepted August 4, 2013.
Communicated by Hsin-Hsi Chen.
* This work is supported by the Natural Science Foundation of China (61105072), the Chinese National Program on Key Basic Research Project (2013CB329304), the Tianjin Younger Natural Science Foundation (14JCQNJC00400) and Tianjin Key Laboratory of Cognitive Computing and Application. The primary work was supported by the NLC group of Microsoft Research Asia and is completed in the Open University, UK.

based on measures of string similarity are widely used [4]. BLEU and NIST [5] compare MT output with references and generate a similarity score. BLEU can also separately reveal matching degrees of phrases of different lengths. This feature provides more details about performance, but is highly limited and difficult to explain. Other methods are based on word alignment. TER [6] improves the edit distance with costs for shifts. CDER [7] enables the re-ordering of blocks. METEOR identifies the alignment of single words. GTM [8] calculates the precision and recall using the longest common subsequence. ROUGE is based on skip-bigram.

Aside the plain words, linguistic features are also helpful in MT evaluation. Some approaches use character-level features [9], some approaches use paraphrasing [10-12], while others use syntax information [13-15]. In these methods, the linguistic features are used only to measure general similarity, not to classify evaluated units.

Some semi-automatic diagnostic evaluation has been proposed [16]. Different from semi-automatic methods, Farrús [17], Popovic [18], and Zhou [19] proposed automatic diagnostic evaluation on target or source languages, respectively.

The general evaluation metrics cannot shed light on the inherent factors of MT system. Developers require an automatic diagnostic evaluation to provide rich information which guides the direction of development. The current manual and semi-automatic diagnostic evaluation acquire considerable manual work, and the automatic diagnostic metric does not aim to analyze the MT systems on bilingual aspects.

The requirement of the diagnostic evaluation is linguistically motivated which can be characterized by four features: (1) Dividing the candidate translations into sub-units; (2) Sub-units being linguistically classified; (3) Sub-units being separately evaluated; (4) Providing the scores of the linguistic categories for the sub-unit classifications.

Yu [20] proposed a special semi-automatic evaluation called MTE. MTE is based on small-size human-crafted checkpoints for English-Chinese MT systems. A checkpoint is a manually constructed linguistic unit composed of source words and human translation, which are instances from six linguistic categories. MTE separately evaluates the checkpoints and merges the scores into an overall score. Because MTE provides only an overall score, it is not a diagnostic metric, but the initial concept of checkpoints remains useful in the diagnostic evaluation of MT systems.

Extending the ideas of automatic general evaluation and manual diagnostic evaluation, we propose a diagnostic evaluation **Woodpecker** which can evaluate the capability of the system to handle important linguistic test cases called **checkpoints**. A checkpoint is composed of a **linguistic unit** and its **references.** A linguistic unit is a linguistically motivated unit in source or target language. The diagnostic evaluation is performed by matching the output of MT systems to the checkpoint references. Extracting checkpoints is an automatic process, in which a word aligner and parsers are used. Woodpecker generates the scores of the checkpoints, categories, and category groups, which can effectively facilitate the diagnostic evaluation of MT systems.

The rest of the paper is structured as follows. Section 2 overviews the Woodpecker with the definition, terminology and category taxonomy; Section 3 presents the automatic construction of the checkpoints; Section 4 illustrates the diagnostic evaluation; Section 5 shows experimental results on MT systems; Section 6 uses the diagnostic scores to improve the ranking of MT systems; Section 7 introduces the utilization of Woodpecker in real tasks; Section 8 provides essential discussions; Section 9 concludes this work.

2. OVERVIEW OF WOODPECKER

In Woodpecker, we first define a taxonomy that contains linguistic categories for evaluation. Then checkpoint database is automatically extracted from bilingual corpus and classified with linguistic categories. This process involves the following steps: (1) Parallel sentences are collected; (2) The source and target sentences are parsed; (3) The linguistic units are identified from the parsed sentences; (4) Word alignment between sentence pairs is performed; (5) For linguistic units on source side, references are determined as corresponding part in the target sentences; (6) Linguistic units and references are constructed into checkpoints.

With the checkpoint database, the diagnostic evaluation is performed with the following steps: (1) Source sentences containing the checkpoints are translated using the candidate MT system; (2) For each checkpoint, the credit conferred to the MT system is calculated by the *n*-grams matching degree against the MT translation; (3) The credit of a category, a category group or the MT system can be obtained by summing up the credits of all the checkpoints belonging to that category, category group or the whole system.

The core conceptions and terms used in Woodpecker are explained as following.

Category: A category in Woodpecker pertains to a lowermost linguistically motivated class that is not divided into subclasses (*e.g.*, nouns, verb phrases, and passive voice sentences). We use *Ca* to denote a specific category.

Category Group: A category group is a collection of categories, denoted by *Gr*. We define two kinds of category groups: **Default groups** are predefined in the system with classical linguistic meanings; **Custom groups** are defined by users.

Checkpoint: A checkpoint is composed of a **linguistic unit** and its **reference(s)**, denoted by *Cp*. A linguistic unit *L* is a sequence of linguistically motivated words defined by a 2-tuple, where *S* denotes the word sequence and *C* is the specific category:

$$L = <S, C>. \tag{1}$$

The reference(s) of a linguistic unit are denoted by $\{r\}$. If *S* is a word sequence of the source language, $\{r\}$ will be a set of correct translations of *S*; if *S* is a word sequence of the target language, $\{r\}$ will be *S* itself. With a linguistic unit $<S, C>$ and its references set $\{r\}$, we use a 3-tuple to present a checkpoint *Cp*:

$$Cp = <S, \{r\}, C>. \tag{2}$$

Checkpoint Collection of a Category or a Category Group: The checkpoint collection of a category is a collection of checkpoints that share the same category *Ca*. Moreover, the checkpoint collection of a category group is a collection of checkpoints whose category belongs to the same category group. We use *Col* to represent a checkpoint collection. If *Col* is a checkpoint collection of category *Ca*, then

$$Col = \{Cp \mid Cp.C = Ca\}. \tag{3}$$

If *Col* is a checkpoint collection of category group *Gr*, then

$$Col = \{Cp \mid Cp.C \in Gr\}. \quad (4)$$

In Woodpecker, the taxonomy of the categories is composed of categories and category groups defined with multi-source linguistic knowledge and four principles: (1) They should be linguistically motivated; (2) They should be widely accepted; (3) They should contain both traditional linguistic categories and extra categories that are important to the translation task; (4) They should be automatically processed within the capability of current Natural Language Process (NLP) tools. In light of these principles, the taxonomy is constructed with following steps: (1) We first adopt the manual Chinese taxonomy in [21, 22] and the manual English taxonomy in [23] together with Wikipedia; (2) Then, we remove the categories that are beyond the tagging set of parsers; (3) Additional important categories are added to the taxonomy; (4) Finally, default category groups that pertain to the same manual taxonomies.

The category taxonomy is constructed into a tree according to the hyponymy relationships between categories. The final taxonomy includes typical categories on three linguistic levels: word, phrase, and sentence. The categories can also be classified according to the types of used linguistic knowledge: (1) Constituent tag; (2) Dependency tag; (3) Dictionary; (4) Manual rules.

We take Chinese-English translation as an example containing 22 Chinese categories and 21 English categories listed in Table 1. Examples of the categories are provided in Table 2, associated with the used linguistic knowledge and example checkpoint.

Table 1. Linguistic category taxonomy of Chinese and English.

Chinese			English		
Word level			Word level		
Ambiguous word	New word	Idiom	Noun	Verb (with tense)	Modal verb
Overlapping word	Collocation	Noun	Adjective	Adverb	Pronoun
Verb	Adjective	Adverb	Preposition	Ambiguous word	Plurality
Pronoun	Preposition	Quantifier	Possessive	Comparative and superlative degree	
Phrase level			Phrase level		
Subject-predicate phrase	Predicate-object phrase	Preposition-object phrase	Noun phrase	Verb phrase	Adjective phrase
Measure phrase	Location phrase		Adverb phrase	Preposition phrase	
Sentence level			Sentence level		
BA sentence[2]	BEI sentence[3]	SHI sentence	Time clause	Reason clause	Condition clause
YOU sentence	Compound sentence		Result clause	Purpose clause	

Table 2. Examples of categories in three linguistic levels, used knowledge and checkpoints.

Category	Related knowledge	Example checkpoints
Word level		
Chinese preposition	Constituent tag	<于, {in} >
Ambiguous English word	Dictionary	<play, {打}>
Phrase level		
Chinese subject-predicate phrase	Dependency tag	< {他*说}, he*said >
English verb phrase	Constituent tag	<play football, {踢足球} >
Sentence level		
Chinese “BA” sentence	Manual rules	<他把(BA)书拿走了., {He took away the book.}
English time clause	Manual rules	<We met after the meeting. , {我们会后见的面}

3. CONSTRUCTION OF CHECKPOINT DATABASE

In the checkpoint database, each sentence pair is associated with the checkpoints of various categories in it. The database is saved in XML format. The following XML segment is an example of a sentence pair and a part of its associated checkpoints:

<Src>他 喜欢 打 篮球</Src> *<Ref>He likes to play basketball</Ref>*
<C:S:Verb><CP><L>2=2</L><R>play</R></CP></C:S:Verb>
<C:T:Verb><CP><L>1=1</L><R>like</R></CP><CP><L>3=3</L><R>play</R></CP></C:T:Verb>

where tag "*<Src>*" indicates the source sentence and "*<Ref>*" indicates the reference(s). "*<C:S:Verb>*" indicates checkpoints of verb on source side ("*S:*"); "*<C:T:Verb>*" indicates checkpoints of verb on target side ("*T:*"). "*<CP>*" indicates a checkpoint. "*<L></L>*" indicates the index range of the words sequence of the linguistic unit. "*<R></R>*" indicates the references of the checkpoints.

Fig. 1 illustrates the framework of the checkpoints extraction. "Linguistic parser" could be POS-tag parser, dependency parser, constituent parser or other automatic linguistic parsers. "Special words dictionaries" could be ambiguous words dictionary or other dictionaries. "Linguistics rules" could be artificial rules to automatically identify special linguistic patterns. Term "segments" include words, phrases and sentences.

Here we introduce the process of extraction with an example. Suppose we have a Chinese-English sentence pairs from a bilingual corpus:

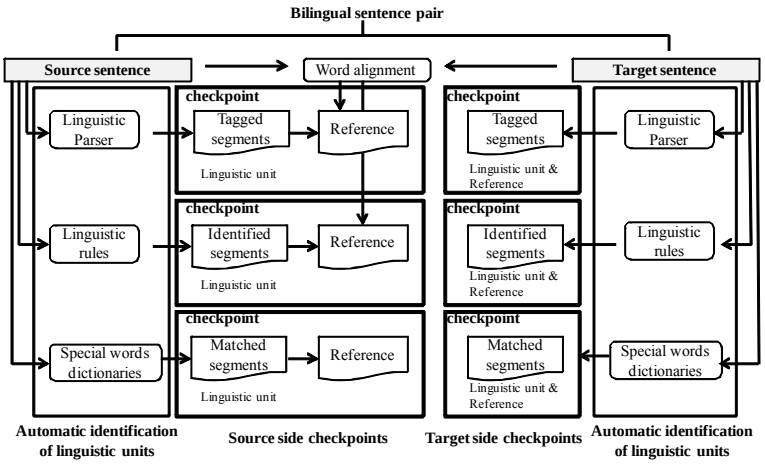


Fig. 1. Main framework of automatic checkpoints extraction.

- **A Chinese-English sentence pair:**
  他 喜欢 打 篮球 *He likes to play basketball*

The following process is an example to extract checkpoints from this sentence pair:

- **Word segmentation and POS-tagging of the source and target sentences:**
  他/PN 喜欢/VV 打/VV 篮球/NN; *He/PRP likes/VBZ to/IN play/VB basketball/NN*

- **Parsing result:**
  SUBJ (*喜欢*, *他*), *COMP*(*喜欢*, *打*), *OBJ*(*打*, *篮球*)
  (*S* (*NP* (*PRP He*)) (*VP* (*VBZ likes*) (*IN to*) (*VP* (*VB play*) (*NP* (*NN basketball*))))))
- **Ambiguous words dictionary matching (e.g., using Hownet [24]):**
  Matched words: 打
- **Identified linguistic units:**
  Source side: <他, Chinese Pronoun>; <喜欢, Chinese verb>; <打, Chinese verb>; <打, ambiguous Chinese word>; <篮球, Chinese noun>; <他喜欢, Chinese subject-predicate phrase>
  Target side: <He, English Pronoun>; <like, English verb>; <to, English Preposition>; <play, English verb>; <basketball, English noun>;<play, ambiguous English word>; <likes to play basket ball, English verb phrase>; <play basket ball, English verb phrase> (one word phrases are omitted)
- **Alignment between source and target words:** (0; 0); (1; 1); (2; 3); (3; 4);

Table 3. Examples of checkpoint extraction.

S	{r}	C	Knowledge used
篮球	{basketball}	Chinese noun	POS tag
打	{play}	Ambiguous Chinese word	Ambiguous word dictionary
他喜欢	{He likes}	Chinese subject-predicate phrase	Dependency parsing tag
play basketball	{play basketball}	English verb phrase	Constituent parsing tag

Table 3 illustrates four typical examples of extracted checkpoints:

Checkpoint "<篮球, {basketball}, Chinese noun>" is extracted with the POS tag. In the source sentence, words "篮球" is tagged as "NN". Then a linguistic unit "篮球" is identified whose category is "Chinese noun". With the alignment information, the corresponding target words of "篮球" is "basketball", which is determined as the reference.

Checkpoint "<打, {play}, Ambiguous Chinese word>" is extracted with the ambiguous word dictionary. Matching the source sentence and the dictionary entries, "打" is found as a Chinese ambiguous word. Then a linguistic unit "打" is identified as "Ambiguous Chinese word". Target word "play" exists both in the dictionary entries of "打" and the target sentence, therefore "play" is determined as the reference.

Checkpoint "<他喜欢, {play basketball}, Chinese subject-predicate phrase>" is extracted with the dependency parsing tag. Words segment "他喜欢" is tagged as "SUBJ" by dependency parser. Then a linguistic unit "他喜欢" is identified whose category is "Chinese subject-predicate phrase". Target phrase "play basketball" is then determined as the reference with the alignment information.

Checkpoint "<play basketball, {play basketball}, English verb phrase>" is extracted with the Constituent parsing tag. Words segment "play basketball" is tagged as "VP" by constituent parser. Then a linguistic unit "play basketball" is identified whose category is "English verb phrase".

3.1 Identifying Linguistic Units

For the categories related to the tags of the constituent tree or dependency structure, the linguistic units are automatically identified as the words with certain tags. Given a

tag, a linguistic unit *L* is identified: the linguistic category of the tag is identified as *L.C*. The word subsequence covered by the tag is identified as *L.S*.

For the categories related to special dictionaries, the words matched by the dictionary entries are identified. Given a matched entry in the source sentence, a linguistic unit *L* is identified: the corresponding category of the dictionary is identified as *L.C*. The word matched by the entry is identified as *L.S*.

For the categories related to the manual rules, predefined rules are used to identify special types of sentences. Given a sentence of special type, a linguistic unit *L* is identified: the type of sentence is identified as *L.C*. The entire sentence is identified as *L.S*.

In Woodpecker default settings, Stanford parser [25], Berkeley parser [25], a Chinese dependency parser [26], WordNet [27], HowNet, a Chinese idiom dictionary and a Chinese new word dictionary are used for syntax parsing and dictionary matching.

3.2 Determining References and Constructing Checkpoints

For a linguistic unit on the target side, the reference is itself. For a linguistic unit on the source side, the determination can be performed with different methods: (1) For the units that are identified by the parser tags, the alignment matrix is used to determine the corresponding part in the target sentences as references; (2) For the special words identified by the dictionary entries, the translations in the dictionaries and target sentence simultaneously are identified as references; (3) For the special type of sentences, the entire target sentences are determined as references.

With each linguistic unit and its reference(s), we construct a checkpoint. Suppose *L:* <*S*, *C*> is an identified linguistic unit, {*r*} is the set of determined references of *L*. A checkpoint *Cp :* <*S*, {*r*}, *C*> can be constructed with *L* and {*r*}:

$$Cp.S = L.S \; ; \; Cp.\{r\} = \{r\} \; ; \; Cp.C = L.C. \tag{5}$$

3.3 Reducing the Noise of the Parser and Alignment

The reliability of linguistic unit primarily depends on the precision of the parsers. We can derive high-quality word level and sentence-level. But some kinds of phrase-level linguistic units cannot be produced by the parsers with high precision. In Woodpecker, the precision is more important than recall for linguistic units identification. Therefore, we select the linguistic units that can be identified by multiple parsers. Table 4 shows the improvement. Columns 2 and 3 show the precision of 6 major types of Chinese phrases of Stanford and Berkeley parsers. Column 4 shows the precision of intersection and column 5 shows the number reduction of linguistic units adopting the intersection. The test corpus is from the Penn Chinese Treebank excluding the training set of the parsers. As shown in Table 4, the precision of the intersection is better in most cases.

Table 4. Precision of the parsers and their intersections.

Phrase Type	Stanford%	Berkeley%	Inter%	Unit# reduction%
Noun Phrase	87.37	86.03	95.83	17.06
Verb phrase	87.34	82.87	95.23	19.68
Preposition Phrase	90.60	88.56	96.00	11.50
Quantifier Phrase	98.12	92.90	99.21	6.31
Adjective Phrase	91.95	90.87	96.41	10.20
Adverb Phrase	95.21	94.25	92.64	3.92

The quality of the references on target side depends on the word alignment accuracy. To reduce noise in the aligner, we use extra knowledge from lexical dictionary to check the reliability of the reference(s). For each reference *r* of a checkpoint *Cp*, we calculate dictionary matching degree *DM*(*r*) with source side (*S*) of *Cp*:

$$DM(r)=\begin{cases} Max\{0.1,\frac{CoCnt(r,Dic(S))}{WordCnt(r)}\} & \text{if } r \text{ is got by word alignment} \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

where *Dic*(*x*) is a word bag containing all the words in the dictionary translations of each source word in *x*. *CoCnt*(*x*, *y*) denotes the number of words common to *x* and *y*. *WordCnt* (*x*) is the number of words in *x*. Specially, if *r* is not obtained based on alignment, *DM*(*r*) will be 1. Because of the limitations of the dictionary, a zero *DM* score does not always indicate an erroneous reference. Thus, we manipulate the *DM* score so that it does not go lower than a specified minimum value (0.1 in Woodpecker).

4. DIAGNOSTIC EVALUATION BY CHECKPOINTS MATCHING

The diagnostic evaluation is performed by matching the references of the checkpoints with the candidate MT translations. To calculate the credit at different occasions of matching, we split the checkpoint reference into a set of *n*-grams and sum up the gains over each gram as the credit for this checkpoint. Additionally, to match word inflections, four different options of matching granularity are defined: (1) Normal: matching with original form; (2) Lower-case: matching with lowercase; (3) Stem: matching with the stem of the word; (4) sense: matching with synonyms.

The procedure for computing the matching degree between the references of checkpoints and the MT translations can be illustrated by Eq. (7). Given a checkpoint *Cp*, and candidate translation *t*, the matching score "*Score*(*Cp*)" between *t* and *Cp* is

$$Score(Cp) = Recall(Cp)*Penalty, \quad (7)$$

where function *Recall*() computes the recall of *n*-gram matching between *t* and the references of *Cp* according to the Eq. (8). Function *Penalty* is used to penalize the redundant *n*-grams in the candidate translation. It penalizes the ratio of average sentence length over the entire reference set *R* and translation set *T*, as shown in Eq. (9):

$$Recall(Cp)=\frac{\sum_{n-gram\in G(r^*)} Match(n-gram)}{\sum_{n-gram\in G(r^*)} Count(n-gram)}, \quad (8)$$

$$Penalty=\begin{cases} \frac{length(R)}{length(T)} & \text{if } length(T)>length(R) \\ 1 & \text{otherwise} \end{cases}. \quad (9)$$

In Eq. (8), *r** is the best reference of *Cp*, which is determined by Eq. (10); *G*(*r*) represents the set of *n*-grams generated by reference *r*. Function *Count*() denotes the total number of *n*-grams; function *Match*() captures the number of *n*-grams occurring in translation *t*; that is, the matching count is calculated.

$$\boldsymbol{r}^{*} = \arg\max_{\boldsymbol{r}\in\{\boldsymbol{r}\}}(\boldsymbol{DM}(\boldsymbol{r}) \times \frac{\sum_{\boldsymbol{n-gram}\in\boldsymbol{G}(\boldsymbol{r})} \boldsymbol{Match}(\boldsymbol{n-gram})}{\sum_{\boldsymbol{n-gram}\in\boldsymbol{G}(\boldsymbol{r})} \boldsymbol{Count}(\boldsymbol{n-gram})}) \tag{10}$$

Aside from the score of a single checkpoint, that of a category or a category group can also be calculated. For category *Ca* or category group *Gr*, the matching score is calculated using Eq. (7), in which *Cp* is substituted with *Ca* or *Gr*. The recall of *Ca* or *Gr* is calculated as the recall of their checkpoint collection, respectively. Giving a checkpoint collection *Col*, the recall of *Col* is determined by Eq. (11), which is a variation of Eq. (8). The *DM* score of each reference is also multiplied to assign weight to the reliability of the checkpoints. *R** is the set of best reference *r** of each checkpoint in *Col*.

$$Recall(Col) = \frac{\sum_{r\in R^{*}} DM(r) \times \sum_{n-gram\in G(r)} Match(n-gram)}{\sum_{r'\in R^{*}} DM(r') \times \sum_{n-gram'\in G(r')} Count(n-gram')} \tag{11}$$

5. EXPERIMENTS ON DIAGNOSTIC EVALUATION

We perform diagnostic evaluation on three SMT systems and a Rule-based MT (RMT) system. The checkpoints database is built with NIST05 test data. Among the three SMT systems, system A is an implementation of classical phrase-based SMT [28]. System B introduces a preprocess to system A to reorder the long source phrases [29]. System C and the RMT system (system D) are external MT systems.

In the first experiment, we diagnosed systems A and B. Results of Chinese categories are selected to be compared in Table 5. According to the BLEU scores, system B

Table 5. Diagnosis of SMT systems with source side (Chinese) checkpoints.

Category	System A	System B	TScore	Variance (A/B)	95% Confidence intervals (A/B)	Count
WORDs						
Ambiguous word	0.5943	0.5957	2.88	0.0099/0.0093	0.5792-0.6095/0.5752-0.6120	3571
New word	0.1842	0.1974	5.56	0.0335/0.0328	0.1496-0.2502/0.1566-0.2634	41
Idiom	0.1933	0.2370	13.38	0.0449/0.0462	0.1353-0.2719/0.1672-0.3421	35
Noun	0.5950	0.5969	2.68	0.0071/0.0067	0.5844-0.6067/0.5856-0.6096	8887
Verb	0.5138	0.5070	–9.41	0.0085/0.0087	0.4994-0.5287/0.4940-0.5193	4502
Adjective	0.5836	0.5577	–17.43	0.0199/0.0205	0.5525-0.6220/0.5299-0.6035	764
Pronoun	0.7566	0.7344	–13.49	0.0241/0.0230	0.7169-0.7952/0.6995-0.7668	576
Adverb	0.5365	0.5433	7.11	0.0150/0.0151	0.5140-0.5633/0.5188-0.5663	1441
Preposition	0.6529	0.6456	–6.21	0.0183/0.0171	0.6246-0.6831/0.6136-0.6826	941
Quantifier	0.5853	0.5737	–4.68	0.0278/0.0273	0.5411-0.6362/0.5391-0.6310	469
Overlapping word	0.3333	0.3958	9.86	0.1028/0.0866	0.1967-0.4482/0.2500-0.5569	25
Collocation	0.6680	0.6598	–8.07	0.0127/0.0132	0.6492-0.6891/0.6280-0.6791	954
PHRASEs						
Subject-predicate phrase	0.5117	0.5176	7.36	0.0134/0.0122	0.4906-0.5318/0.4983-0.5446	525
Predicate-object phrase	0.4041	0.4180	15.52	0.0108/0.0124	0.3857-0.4222/0.4005 -0.4342	1276
Preposition-object phrase	0.4409	0.5125	9.51	0.0143/0.0140	0.4122-0.4905/0.4763-0.5519	125
Measure phrase	0.5030	0.5092	3.56	0.0145/0.0139	0.4794-0.5279/0.4942-0.5295	893
Location phrase	0.5245	0.5338	2.83	0.0343/0.0448	0.4752-0.5835/0.4869-0.5892	56
GROUPs						
WORDs	0.4739	0.4875	8.03	0.0125/0.0118	0.4492-0.4946/0.4596-0.5041	22206
PHRASEs	0.4744	0.4964	13.97	0.0139/0.0155	0.4525-0.4942/0.4723-0.5213	2875
SYSTEM						
SYSTEM (Linguistic)	0.4263	0.4370	16.50	0.0069/0.0078	0.4160-0.4474/0.4225-0.4461	26496
SYSTEM (BLEU)	0.3564	0.3614	7.91	0.0067/0.0061	0.3488-0.3704/0.3536-0.3730	

Table 6. Diagnosis of SMT & RMT systems. Scores are listed as "SMT score/RMT score".

Category	Normal	Lowercase	Stem	Sense
SMT significantly leads categories				
Source (Chinese) categories				
Ambiguous word	0.49 / 0.42	0.50 / 0.42	0.53 / 0.46	0.63 / 0.55
Quantifier	0.75 / 0.70	0.79 / 0.74	0.79 / 0.74	0.79 / 0.74
Collocation	0.66 / 0.54	0.66 / 0.55	0.70 / 0.56	0.73 / 0.60
Subject-predicate phrase	0.46 / 0.30	0.51 / 0.36	0.58 / 0.42	0.63 / 0.45
Predicate-object phrase	0.37 / 0.25	0.37 / 0.26	0.47 / 0.29	0.50 / 0.30
Target (English) categories				
Noun	0.39 / 0.34	0.40 / 0.34	0.43 / 0.35	0.51 / 0.42
Ambiguous word	0.50 / 0.41	0.50 / 0.42	0.55 / 0.43	0.65 / 0.54
Adjective	0.52 / 0.47	0.53 / 0.47	0.56 / 0.49	0.64 / 0.53
Noun phrase	0.37 / 0.31	0.39 / 0.32	0.41 / 0.34	0.47 / 0.40
Adjective phrase	0.47 / 0.39	0.51 / 0.41	0.53 / 0.44	0.61 / 0.49
RMT significantly leads categories				
Source (Chinese) categories				
Idiom	0.43 / 0.66	0.46 / 0.67	0.51 / 0.71	0.57 / 0.76
Pronoun	0.60 / 0.68	0.69 / 0.75	0.66 / 0.75	0.66 / 0.75
Target (English) categories				
Modal verb	0.49 / 0.54	0.49 / 0.54	0.49 / 0.54	0.49 / 0.54
Pronoun	0.60 / 0.68	0.69 / 0.75	0.66 / 0.75	0.66 / 0.75

achieves a non-significant 0.5 percentage points increase to system A. The diagnostic results provide richer information indicating that the two systems perform in a similar manner at word level, but in all phrase-level categories, system B performs better. This result reflects the benefit of reordering complex phrases. The paired t-statistic scores (denoted by 'T score') are calculated following [30]. All the differences are significant (> 95%). The last column shows the number of the checkpoints of each category.

In the second experiment, we diagnosed systems C and D. The BLEU scores of C and D are 0.3005 and 0.2606. In Table 6, the categories are divided into two groups: the categories in which SMT significantly leads and the categories in which RMT significantly leads. Only the words and phrase level categories of significant difference are selected. The diagnostic scores are calculated with the four matching options. The results indicate that system C performs better on most categories than system D, but system D performs better on categories such as idioms, pronouns and modal verb. This result reveals that SMT works well on open categories handling by context, whereas RMT works well on closed categories that can be easily translated by rules or dictionaries.

6. EXPERIMENTS ON RANKING MT SYSTEMS

In [19], the evaluation was alternately formulated as a ranking problem. The experiments demonstrated that the dependency matching rate can increase ranking accuracy in some cases. Compared with dependency structures, the linguistic categories in Woodpecker are more extensive. In the experiments, we used the scores of linguistic categories, dependency matching rates, BLEU scores (v.11b+default settings), and other popular metrics as ranking features of MT systems and trained by Ranking SVM of SVMlight [31], on LDC2006T04 [32], ranking 7 MT translations with three-fold cross-validation on sentence and document level. Spearman ρ was used to calculate the correlation with

Table 7. Ranking of MT translations with different feature sets at sentence level.

Sentence level		
Metric	Mean co-rrelation	95% confidence intervals
BLEU 4	0.245#	0.241 – 0.256
DP	0.246#	0.230 – 0.266
GTM (e=2)	0.251#	0.246 – 0.260
LC	0.263#	0.247 – 0.267
BLEU+DP	0.270#	0.265 – 0.281
BLEU+ LC	0.288#	0.273 – 0.294
METEOR(exact)	0.306#	0.288 – 0.312
NIST 5	0.307#	0.286 – 0.312
BLEU+ DP +LC	0.307#	0.289 – 0.320
NIST+ LC	0.322*	0.309 – 0.340
METEOR(exact&syn)	0.327*	0.321 – 0.345
NIST+ DP +LC	0.333*	0.317 – 0.350

Table 8. Ranking of MT translations with different feature sets at document level.

Sentence level		
Metric	Mean co-rrelation	95% confidence intervals
BLEU 4	0.305#	0.295 – 0.311
DP	0.323*	0.310 – 0.326
GTM (e=2)	0.325*	0.316 – 0.332
LC	0.327*	0.324 – 0.331
BLEU+DP	0.332*	0.321 – 0.338
BLEU+ LC	0.359*	0.349 – 0.363
METEOR(exact)	0.363*	0.355 – 0.372
NIST 5	0.369*	0.352 – 0.376
BLEU+ DP +LC	0.373*	0.366 – 0.379
NIST+ LC	0.387*	0.372 – 0.399
METEOR(exact&syn)	0.394*	0.388 – 0.401
NIST+ DP +LC	0.409*	0.400 – 0.412

human assessments. Tables 7 and 8 show the results of different feature sets. "DP" denotes dependency matching rate; "LC" denotes linguistic categories. Marks '*' and '#' indicate that the correlation is significant at higher than 99% and 95% levels, respectively. The 95% confidence intervals of the correlations are also given.

In the results, the LC, when used alone, are better related with human assessments than BLEU and GTM. When combined with BLEU and NIST, LC scores improve the correlation score. This improvement is better than that made by DP. The combination of NIST+LC is better than that of METEOR (exact) at sentence and document level, and METEOR (exact&syn; syn denotes the synonym module) at document level. The results indicate the ability of linguistic features to improve ranking performance.

7. ANALYSIS IN OPEN EVALUATION TRACKS

Woodpecker Toolkit is distributed by Microsoft Research and was selected as one of the criteria in the evaluation tracks of China Workshop of Machine Translation (CWMT) 2008-2013. CWMT tracks are held by the Institute of Computing Technology, Chinese Academy of Sciences [33, 34]. CWMT introduced the Woodpecker into its tracks to examine MT performance on linguistic phenomena, together with the mainstream metrics such as BLEU, NIST and GTM. Here we introduce the analysis of typical diagnostic results of Woodpecker in CWMT supported by [33].

The introduced CWMT evaluation has four tracks, as shown in Table 9. "System Combination" is a track to combine the outputs of MT systems to obtain better translations. There are 15 participants of the tracks. During the diagnosis, CWMT extended the linguistic taxonomy to 27 Chinese categories and 30 English categories. GIZA++ was used for word alignment, Stanford Parser was used for dependency and constituent structures and Berkeley Parser was used for constituent structure.

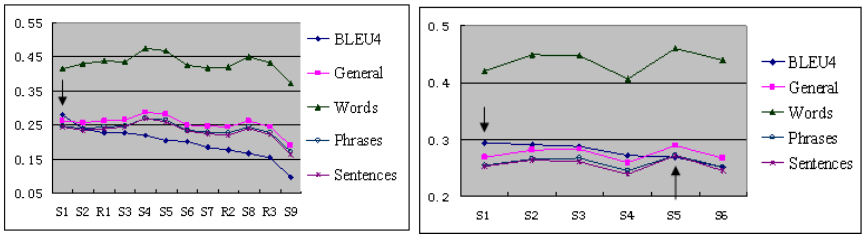
7.1 Correlations between Scores of Woodpecker and BLEU

Figs. 2 and 3 show the BLEU score, the Woodpecker's general score, and the scores of word, phrase and sentence level in three tasks. The horizontal axes indicate the dif-

ferent MT systems, where 'S' indicates SMT and 'R' indicates RMT. The vertical axes indicate the scores. The correlations between BLEU and the Woodpecker scores of all systems were also calculated with Spearman and Pearson coefficients in Table 10.

Table 9. Evaluation tracks. S&T = Scientific and technical literature.

Language	Domain	Task
Chinese to English	News	Machine Translation
Chinese to English	News	System Combination
English to Chinese	News	Machine Translation
English to Chinese	S&T	Machine Translation



(a) News translation task. (b) Combination task.

Fig. 2. Scores of woodpecker and BLEU in C2E task.

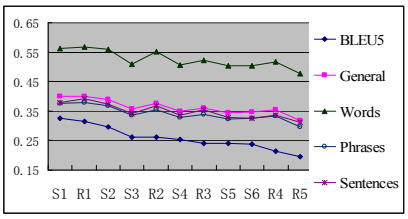


Fig. 3. Scores of woodpecker and BLEU in E2C news translation task.

Table 10. Correlations between Woodpecker and BLEU. *N* is the number of MT systems.

Task	*N*	Score	Spearman	Pearson
C2E_news	12	General	0.5594	0.7182
		Words	0.1678	0.4138
		Phrases	0.6224	0.7425
		Sentences	0.6923	0.7607
C2E_news_combine	6	General	0.0857	0.2232
		Words	-0.1429	-0.0828
		Phrases	0.0857	0.2682
		Sentences	0.1429	0.3263
E2C_news	11	General	0.8727	0.9283
		Words	0.8273	0.8747
		Phrases	0.8727	0.9340
		Sentences	0.9182	0.8929
E2C_S&T	9	General	0.8500	0.9429
		Words	0.8333	0.9452
		Phrases	0.8500	0.9421
		Sentences	0.8167	0.9186

From Table 10, good correlation can be observed between scores of the Woodpecker and BLEU in most cases. The correlation for "C2E-news" task is lower than those for "E2C-news" and "E2C-S&T(Scientific and Technical Literature)" tasks. One

Table 11. Comparison of five systems with human evaluation.

Metrics		S1	R1	S2	S-Combine1	S-Combine5
NIST-BLEU		0.2809	0.2275	0.2264	0.2944	0.2679
IBM-BLEU		0.2661	0.2215	0.2137	0.2792	0.2588
BLEU-SBP		0.2631	0.2193	0.2122	0.2758	0.2560
Wood-pecker scores	G	0.2629	0.2618	0.2649	0.2686	0.2887
	W	0.4146	0.4377	0.4354	0.4209	0.4608
	P	0.2480	0.2446	0.2482	0.2536	0.2719
	S	0.2459	0.2401	0.2457	0.2519	0.2709
Human evaluation	F	3.4	3.45	3.2	3.2	3.45
	A	3	3.7	3.25	3	3.6

possible reason is that the SMT system with the highest BLEU score in "C2E_news" task has low Woodpecker scores (S1 in Fig. 2 (a), marked with arrow). In"C2E_news_ combine" task, the correlation between Woodpecker and BLEU is destroyed by S1 and S5 (Fig. 2 (b), marked with arrows). These phenomena will be discussed in Section 7.3.

7.3 Case Study with Human Evaluation

S1 with the highest BLEU score has low Woodpecker scores. S1 and S5 in Fig. 2 (b) destroy the correlation between Woodpecker and BLEU. 20 sentences are selected uniformly distributed in the test data of "C2E_news" task as new test data to study these cases. A human evaluation was conducted on the results of these three systems and other two systems: R1 and an S2. We assigned each sentence a subjective 1-5 score along two axes: adequacy and fluency [35]. The results are shown in Table 11, where G=General, W=Words, P=Phrases, S=Sentences, F=Fluency, A=Adequacy.

Based on the results, S1 has the lowest word level score and adequacy score among three MT systems. As to the system combination, except for the three kinds of BLEU scores, all the Woodpecker scores and human evaluation scores of S-Combine1 are lower than those of S-Combine5, especially the adequacy score. We examined the output of S-Combine1, and found its translation sentences to be very similar to those of the S1. The sentence in Fig. 5 is an example. This similarity is because S-Combine1 uses the technique of sentence-level system combination and assigns the Top 1 hypothesis (S1) the highest score. We also examined the system description of S-Combine5, and found that this system uses the word-level system combination technique. Although the BLEU scores are lower, its translation adequacy is much better, as seen in Fig. 4.

From this case study, a significant amount of linguistic information in the test data was lost, despite the fact that the *n*-gram matching between the reference and candidate system translations increased, which led to higher BLEU score. This might be the major reason for the high BLEU score but low adequacy and Woodpecker scores. In this sense, Woodpecker can provide more reasonable measurement of the adequacy, with the results being closer to human evaluation on general and diagnostic levels.

Source: 张秀华家挂的胡、温画像是经过电脑处理，原来画面的其他人员已经被掩盖，只有两个人握手的画面。
S-Combine1: Zhang Xiuhua, a computer processing, and other personnel have been only two people.
S-Combine5: Zhang Xiuhua home hanging on Hu, warm portrait is through the computer processing, so that the other personnel have been covered, there are only two shake hands.
Top 1: Zhang Xiuhua, after computer processing, and other personnel have been only two people.
Top 2:, moustache after the computer , the picture of the other has been masked , only two shook hands .
Top 3: Zhang Xiuhua family hung Hu and Wen Hua like after the computer, the picture of the other has been covered up , only two shook hands .
Reference 1: The portrait of Hu and Wen hung in Zhang Xiuhua's home has been processed by the computer; the other officials present were edited out to show only the two shaking hands.

Fig. 4. Example of System Combination; Top 1-3 are outputs from three single systems with the highest BLEU for combination.

8. DISCUSSION

8.1 Contributions and Limitations

In general, Woodpecker is believed to have two main contributions: First, the fundamental idea of Woodpecker encourage to perform diagnostic evaluation in NLP or AI, and the diagnosis is suggested to be supervised by the rich knowledge. To this direction, Woodpecker proposes a successful case in machine translation. Second, Woodpecker committed to tell the MT researchers and developers about the inherent characters of their systems with little manual work. And then, targeted improvement can be performed, which greatly facilitate the development process.

In addition, Woodpecker also involves contributions in specific methods. Aside from the integration of considerable existing knowledge and technologies, several new efforts are also contributed: (1) Refinement of the classical linguistic system that refers to the requirements for MT diagnosis; (2) Filtering of the syntactic segments identified by the parsers, thereby enhancing the precision of the linguistic units; (3) Validating the word alignments with the dictionary; (4) Applying the *n*-gram metric in a hierarchical structure and providing the scores of different levels.

Woodpecker improves the preliminary idea of evaluating the MT systems with categories [19] to make an integrated diagnostic methodology with rich linguistic knowledge. The conceptions and terminology are defined and formally expressed. Both source and target categories are considered to provide much richer diagnostic results. Woodpecker Toolkit was developed and distributed by Microsoft Research to the public. The Toolkit was used by CWMT open evaluation tracks and thus be verified by third-party results. Besides the improvement in many aspects, there are still some limitations:

One limitation is the demand of the manual linguistic taxonomy. Although the manual taxonomy is well defined, such taxonomy is unavailable in some minor languages. We propose a technology that automatically mines the sub-structures frequently found in the syntactic trees and regards the mined patterns as important linguistic phenomena. The initial experimental results show that the method successfully identifies many known patterns in Chinese [37].

Another limitation is the demand of multiple parsers to improve the precision of the checkpoints. We also propose an alternative to filtering the syntactic segments from single parsers. The method calculates mutual information of certain categories of syntactic segments and their contexts. Then, the segments whose contexts have more substantial information in common with the category are selected. The initial experimental results show that this approach works for six most frequent Chinese phrases [38].

8.2 Comparison with Related Work

Woodpecker is inspired by MTE with many extensions. MTE is based on human craft linguistic taxonomy and checkpoints. The test corpus is static and small. While the test corpus and checkpoints of Woodpecker is fully automatically obtained. And, MTE only use binary scores, while Woodpecker uses *n*-gram matching rates.

N-gram-based methods and Woodpecker differ in many ways. In *n*-gram approaches, a sentence is viewed as a collection of *n*-grams without differentiating linguistic aspects. In Woodpecker, a sentence is viewed as a collection of checkpoints with different

types and depths. Furthermore, *n*-gram approach typically only provides overall score or partially enable fine-grained glimpse into translations. But these fine-grained scores are still used to general evaluation. On the contrary, Woodpecker is designed in full diagnostic style. It provides scores of linguistic categories and provides much richer information, helping developers concretely identify the strengths and flaws of a system.

Woodpecker also differs from current methods that use linguistic features. Other methods only select features that are effective in system-level evaluation. Woodpecker proposes an entire taxonomy of important linguistic categories in different levels.

Compared with semi-automatic diagnostic metrics, Woodpecker proposes an automatic process from checkpoints extraction to diagnostic evaluation. And, compared with the automatic diagnostic metrics with respect to a relatively small set of errors in target language or source language, Woodpecker can analyse the MT systems with an entire linguistic taxonomy on different levels of both target and source languages, and thus provide much richer information about the performance of the MT systems.

9. CONCLUSION

This paper presents an automatic diagnostic evaluation metric for MT systems based on linguistic categories and automatically constructed checkpoints. In contrast to metrics, which provide only overall scores, the proposed metric can provide developers feedback on the weaknesses and strengths of an MT system in relation to specific linguistic categories or category groups. In contrast to existing diagnostic systems that are based on checkpoints, the new metric constructs integrated category taxonomy and presents an approach to automatically generating a checkpoint database on both target and source side. We show that although some noise occurs in word alignment and parsing, we can reduce this problem by refining the parser results, assigning weights to references with confidence scores, and providing numerous checkpoints.

The experiments and application in real task demonstrate that this method can reveal specific differences among MT systems of similar or different architectures. Woodpecker is also promising to provide more reasonable measurement of the adequacy of the translations, with the experimental results being closer to small-sized human evaluation than BLEU, on the general and diagnostic levels. We also demonstrate that the linguistic checkpoints can be used as new features for enhancing the ranking of MT systems.

This work is also a scalable approach. Although we demonstrate the diagnostic evaluation method with Chinese-English translation, our approach can be applied to other language pairs, only if the syntax parser and word aligner are available. The category taxonomy of the system is flexible. Users can freely add new categories and category groups. With the proposed technology of automatic checkpoint construction, researchers can build a similar system based on novel taxonomy and corresponding parsers.

REFERENCES

1. K. Papieni, *etc.*, "BLEU: a method for automatic evaluation of machine translation," in *Proceedings of the Association for Computational Linguistics*, 2002, pp. 311-318.
2. C. Y. Lin and F. J. Och, "Automatic evaluation of machine translation quality using

longest common subsequence and skip-bigram statistics," in *Proceedings of the 42nd Meeting of the Association for Computational Linguistics*, 2004, pp. 605-612.
3. S. Banerjee and A. Lavie, "METEOR: an automatic metric for MT evaluation with improved correlation with human judgements," in *Proceedings of ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 2005, pp. 65-72.
4. C. C. Burch, *etc.*, "Findings of the 2011 workshop on statistical machine translation," in *Proceedings of the 6th Workshop on Statistical Machine Translation*, 2006, pp. 22-64.
5. G. Doddington, "Automatic evaluation of machine translation quality using *n*-gram co-occurrence statistics," in *Proceedings of the 2nd International Conference on Human Language Technology Research*, 2002, pp. 128-132.
6. S. Matthew, *etc.*, "TER-Plus: paraphrase, semantic, and alignment enhancements to translation edit rate," *Machine Translation*, Vol. 23, 2009, pp. 117-127.
7. L. Gregor, *etc.*, "CDER: Efficient MT Evaluation Using Block Movements," in *Proceedings of the European Chapter of the ACL*, 2006, pp. 241-245.
8. I. D. Melamed, *etc.*, "Precision and recall of machine translation," in *Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 2003, pp. 61-63.
9. C. Liu and H. T. Ng, "Character-level machine translation evaluation for languages with ambiguous word boundaries," in *Proceedings of Association for Computational Linguistics*, 2012, pp. 921-929.
10. L. Zhou, *etc.*, "Re-evaluating machine translation results with paraphrase support," in *Proceedings of the Empirical Methods on Natural Language Processing*, 2006, pp. 77-84.
11. D. Kauchak and R. Barzilay, "Paraphrasing for automatic evaluation," in *Proceedings of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 2006, pp. 455-462.
12. K. Owczarzak, *etc.*, "Contextual bitext-derived paraphrases in automatic MT evaluation," in *Proceedings of Association for Machine Translation in the Americas*, 2006, pp. 86-93.
13. E. Amigó, *etc.*, "MT evaluation: human-like vs. human acceptable," in *Proceedings of the Association for Computational Linguistics*, 2006, pp. 17-24.
14. J. Giménez and L. Màrquez, "Linguistic features for automatic evaluation of heterogeneous MT systems," in *Proceedings of Workshop of Statistical Machine Translation in Conjunction with 45th ACL*, 2007, pp. 256-264.
15. Y. Ye, *etc.*, "Sentence level machine translation evaluation as a ranking problem: one step aside from BLEU," in *Proceedings of Workshop of Statistical Machine Translation in Conjunction with 45th ACL*, 2007, pp. 240-247.
16. K. Kirchhoff, *etc.*, "Semi-automatic error analysis for large-scale statistical machine translation," in *Proceedings of Machine Translation Summit*, 2007, pp. 56-63.
17. M. Farrús, *etc.*, "Study and correlation analysis of linguistic, perceptual, and automatic machine translation evaluations," *Journal of the American Society for Information Science and Technology*, Vol. 63, 2011, pp. 174-184.
18. M. Popović, "Class error rates for evaluation of machine translation output," in *Proceedings of the 7th Workshop on Statistical Machine Translation*, 2012, pp. 71-75.

19. M. Zhou, *etc.*, "Diagnostic evaluation of machine translation systems using automatically constructed linguistic checkpoints," in *Proceedings of International Conference on Computational Linguistics*, 2008, pp. 1121-1128.
20. S. Yu, "Automatic evaluation of output quality for machine translation systems," *Machine Translation*, Vol. 8, 1993, pp. 117-126.
21. J. Lv, *Foundation of Mandarin Grammar*, Shangwu Publisher, Beijing, 2000.
22. S. Liu, *Linguistics of Contemporary Chinese Language*, Advanced Education Publisher, Beijing, 2002.
23. R. Huddleston, *Introduction to the Grammar of English*, Cambridge University Press, Cambridge, 1984.
24. D. Klein, *etc.*, "Accurate unlexicalized parsing," in *Proceedings of the Association for Computational Linguistics*, 2003, pp. 423-430.
25. http://www.keenage.com.
26. G. A. Miller, "WordNet: A Lexical database for English," *Communications of the ACM*, Vol. 38, 1995, pp. 39-41.
27. M. Zhou, "A block-based robust dependency parser for unrestricted Chinese text," in *Proceedings of the Chinese Language Processing Workshop*, 2000, pp. 78-84.
28. F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, Vol. 29, 2003, pp. 19-51.
29. D. Zhang, *etc.*, "Phrase reordering model integrating syntactic knowledge for SMT," in *Proceedings of the Empirical Methods on Natural Language Processing*, 2007, pp. 533-540.
30. C. Li, *etc.*, "A probabilistic approach to syntax-based re-ordering for SMT," in *Proceedings of the Association for Computational Linguistics*, 2007, pp. 720-727.
31. P. Koehn, "Statistical significance tests for machine translation evaluation," in *Proceedings of the Empirical Methods on Natural Language Processing*, 2004, pp. 102-109.
32. T. Joachims, "Making large-scale support vector machine learning practical," *Advances in Kernel Methods: Support Vector Machines*, MIT Press, MA, 1998.
33. X. Ma, "Multiple-translation Chinese (MTC) part 4," LDC Catalog No. LDC2006T-04, 2006.
34. H. Zhao, *etc.*, "Introduction to China's CWMT2008 machine translation evaluation," in *Proceedings of MT Summit XII*, 2009, pp. 160-167.
35. H. Zhao, *etc.*, "Summary on CWMT2011 MT translation evaluation," *Journal of Chinese Information Processing*, Vol. 26, 2012, pp. 22-30.
36. LDC, "Linguistic data annotation specification: Assessment of fluency and adequacy in translation," Revision 1.5, 2005.
37. B. Wang, *etc.*, "Stability vs. effectiveness: Improved sentence-level combination of machine translation based on weighted MBR," in *Proceedings of International Conference on Asian Language Processing*, 2009, pp. 39-42.
38. B. Wang, *etc.*, "Discover linguistic patterns in parsed corpus with frequent subtree mining," in *Proceedings of International Conference on Knowledge Discovery and Data Mining*, 2010, pp. 86-89.
39. B. Wang, *etc.*, "Automatic syntactic segment filtration for mass syntax corpus with mutual information," in *Proceedings of International Conference on Future Information Technology and Management Engineering*, 2010, pp. 234-237.

Bo Wang is an Associate Professor at the Computer School, Tianjin University. He got his doctor's degree at Harbin Institute of Technology, China, 2010. His major interests are the machine translation and natural language processing.

Ming Zhou, senior researcher, manager of Microsoft Research Asia Natural Language Computing Group. He is an expert in the areas of machine translation and natural language processing. His Chinese-Japanese machine translation software product was granted Makoto Nagao Award. He is the key inventor and technology leader of the famous AI gaming of Chinese Couplets Generation and the English Assistance Search Engine, Engkoo.

Shujie Liu is a Researcher at the Natural Language Processing Group, Microsoft Research Asia. He got his doctor's degree at the Harbin Institute of Technology, China, 2012. His major interests are the machine translation and natural language processing technologies.

Mu Li is a Senior Researcher at the Natural Language Processing Group, Microsoft Research Asia. He got his doctor's degree at the Northeast University, China, 2001. His major interests are the natural language processing technologies.

Dongdong Zhang is a Researcher at the Natural Language Processing Group, Microsoft Research Asia. He got his doctor's degree at the Harbin Institute of Technology, China, 2005. His major interests are the natural language processing technologies.