Dependency Grammar Based English Subject-Verb Agreement Evaluation

Dongfeng Cai[a], Yonghua Hu[a], Xuelei Miao[a], and Yan Song[b]

[a]Knowledge Engineering Research Center
Shenyang Institute of Aeronautical Engineering
No.37 Daoyi South Avenue, Daoyi Development District, Shenyang China 110136
caidf@vip.163.com, sam830829@hotmail.com, miaoxl@ge-soft.com
[b]Department of Chinese, Translation and Linguistics
City University of Hong Kong
83 Tat Chee Ave, Kowloon, Hong Kong
yansong@student.cityu.edu.hk

Abstract. As a key factor in English grammar checking, subject-verb agreement evaluation plays an important part in assessing translated English texts. In this paper, we propose a hybrid method for subject-verb agreement evaluation on dependency grammars with the processing of phrase syntactic parsing and sentence simplification for subject-verb discovery. Experimental results on patent text show that we achieve an F-score of 91.98% for subject-verb pair recognition, and a precision rate of 97.93% for subject-verb agreement evaluation on correctly recognized pairs in the previous stage.

Keywords: Subject-verb agreement, Sentence simplification, Dependency grammar, phrase syntactic parsing

1 Introduction

Subject-verb agreement error is the most common type of mistakes made in translating other languages to English text, and affects the quality of the generated text considerably. By making a detailed analysis on 300,000 error-noted English patent texts, we found that the subject-verb agreement errors comprise 21.7% of all the translation errors. It is obviously indicated that subject-verb agreement is one of the common problems translators would encounter. Due to the complicate grammar and flexible usage of sentence types, especially the complicated relationship between subjects and predicate verbs, the subject-verb agreement evaluation is a difficult mission to tackle.

Currently, manual proofreading is still the main approach widely applied in detecting subject-verb agreement errors made by translators. However, it costs too much while in low efficiency, and manual work is not capable of reuse. To solve this problem, a computational approach is proposed in this paper to automatically recognize the subject-verb pairs and evaluate their agreement by obtaining the dependency relationship between the subjects and its predicate verbs. Phrase syntactic parsing and sentence simplification are used and proved to be effective in our routine.

The rest of the paper is organized as follows: a concise survey of related works is presented in the next section; section 3 is the description of our method; section 4 illustrates the procedure of our experiments; and the experimental results with analysis are presented in section 5; section 6 is the conclusion.

2 Related Works

For there are limited researches exclusively focus on subject-verb agreement, many related works are reported on dealing with grammatical errors, some of which includes subject-verb agreement case. Atwell (1987), Bigert and Knutsson (2002), Chodorow and Leacock (2000) proposed the n-gram error checking for finding grammatical errors. Hand-crafted error production rules (or “mal-rules”), with context-free grammar, are designed for a writing tutor for deaf students (Michaud et al., 2000). Similar strategies with parse trees are pursued in (Bender et al., 2004), and error templates are utilized in (Heidorn, 2000) for a word processor. An approach combining a hand-crafted context free grammar and stochastic probabilities is proposed in (Lee and Seneff, 2006) for correcting verb form errors, but it is designed for restricted domain. A maximum entropy model using lexical and part of speech(POS) features, is trained in (Izumi et al., 2003) to recognize a variety of errors, and achieves 55% precision and 23% recall on evaluation data. John Lee and Stephanie Seneff (2008) proposed a method based on irregularities in parsing tree and n-gram, to correct English verb form errors made by non-native speakers, and achieved a precision around 83.93%. However, on subject-verb agreement processing, it mainly aimed at those sentences which are relatively simple, and proved some *wh-* subject problems to be difficult for its approach.

3 Our Method

3.1 Research Issues

During the translation process, the subject-verb disagreement phenomenon is common, especially the confusion between the base form and the third person singular form. E.g. the sentence: *the utility model disclose a mosaic thrust bearing shell.* The subject ‘*model*’ and the predicate verb ‘*disclose*’ do not agree with each other. This aparts the sentence from good quality and should be checked in the proofreading process. Sentences that regard subject-verb disagreement errors as the main target are considered here.

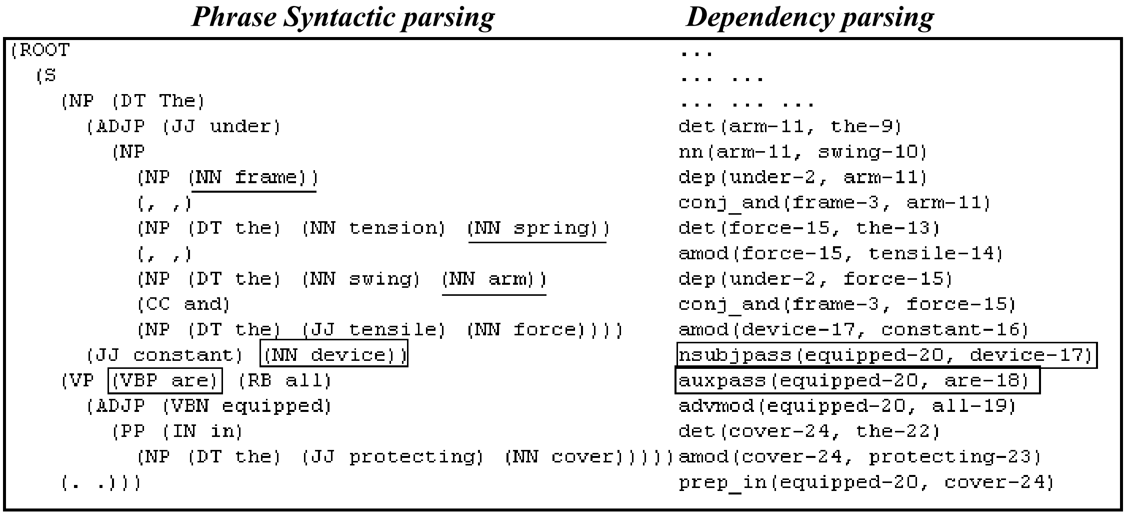
There are many factors involved that can disturb the recognition and agreement evaluation of subject-verb, mainly on semantic level and syntactic level. In detail as follows:

Semantic level It is concerned with inappropriate choices of tense, aspect, voice or mood. E.g., the subject-verb pair recognition is correct, but the verb form does not agree with the context on the semantic level. Such as, *He *ate some bread for his breakfast.* The predicate verb *‘ate’* is in past tense, it agrees with the subject on sentence level. But if its context features need it to be in future tense, the verb form will have to be modified. Here, the checking is only done on syntactic level without considering the context.

Syntactic level As the second type, it can be subdivided into two sub-classes:

(1) Too many modifiers in the sentence may disturb the dependency parsing and phrase syntactic parsing. E.g.,

*The under *frame, the tension *spring, the swing *arm and the tensile force constant *device are all equipped in the protecting cover. Parsed as follows in Figure 1:*



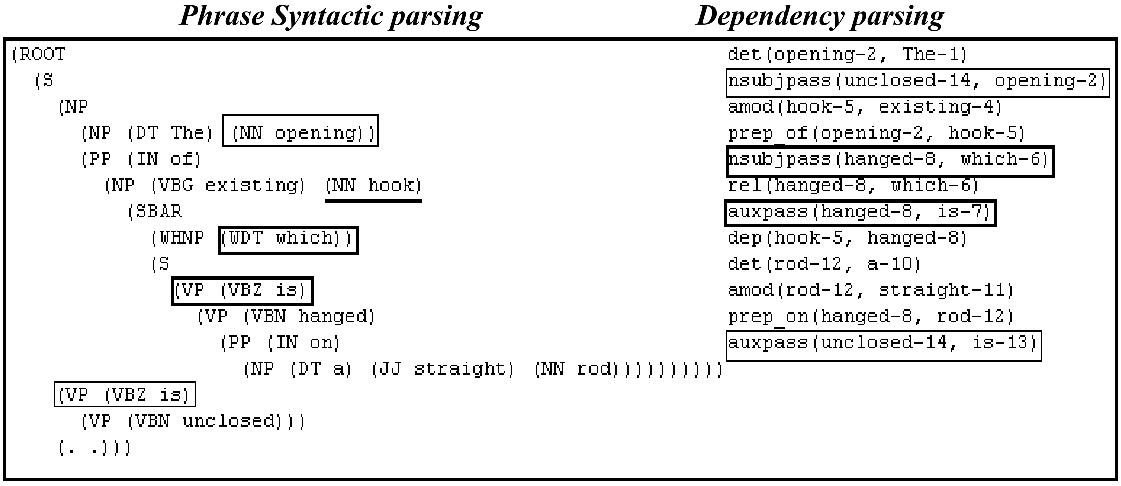
Phrase Syntactic parsing **Dependency parsing**

```
(ROOT                                                   ...
  (S                                                    ... ...
    (NP (DT The)                                        ... ... ...
      (ADJP (JJ under)                                  det(arm-11, the-9)
        (NP                                             nn(arm-11, swing-10)
          (NP (NN frame))                               dep(under-2, arm-11)
          (, ,)                                         conj_and(frame-3, arm-11)
          (NP (DT the) (NN tension) (NN spring))        det(force-15, the-13)
          (, ,)                                         amod(force-15, tensile-14)
          (NP (DT the) (NN swing) (NN arm))             dep(under-2, force-15)
          (CC and)                                      conj_and(frame-3, force-15)
          (NP (DT the) (JJ tensile) (NN force))))       amod(device-17, constant-16)
      (JJ constant) (NN device))                        nsubjpass(equipped-20, device-17)
    (VP (VBP are) (RB all)                              auxpass(equipped-20, are-18)
      (ADJP (VBN equipped)                              advmod(equipped-20, all-19)
        (PP (IN in)                                     det(cover-24, the-22)
          (NP (DT the) (JJ protecting) (NN cover)))))amod(cover-24, protecting-23)
    (. .)))                                             prep_in(equipped-20, cover-24)
```

Figure 1: Example for class 1 on Syntactic level.

In this sentence, '*frame-3*', '*spring-7*', '*arm-11*' and '*device-17*' actually share the same verb '*are-18*'. But as a result of the modifiers such as 'JJ' and 'NN' (Santorini, 1990), the subject is only recognized as '*device-17*' from '*nsubjpass(equipped-20, device-17)*' (de Marneffe et al., 2008.), with other four omitted. As regard to this, sentence simplification is introduced to compress the sentence structure and avoid the disturbance of too many modifiers and some other elements.

(2) The subject-verb pairs have been recognized, but the information that the subject and the predicate verb offer is not enough to evaluate if they are in agreement. E.g.,
*The opening of existing hook *which is hanged on a straight rod is unclosed.*
The sentence contains a *wh-* subordinate clause. The phrase syntactic parsing and dependency parsing are:



Phrase Syntactic parsing **Dependency parsing**

```
(ROOT                                                       det(opening-2, The-1)
  (S                                                        nsubjpass(unclosed-14, opening-2)
    (NP                                                     amod(hook-5, existing-4)
      (NP (DT The) (NN opening))                            prep_of(opening-2, hook-5)
      (PP (IN of)                                           nsubjpass(hanged-8, which-6)
        (NP (VBG existing) (NN hook)                        rel(hanged-8, which-6)
          (SBAR                                             auxpass(hanged-8, is-7)
            (WHNP (WDT which))                              dep(hook-5, hanged-8)
            (S                                              det(rod-12, a-10)
              (VP (VBZ is)                                  amod(rod-12, straight-11)
                (VP (VBN hanged)                            prep_on(hanged-8, rod-12)
                  (PP (IN on)                               auxpass(unclosed-14, is-13)
                    (NP (DT a) (JJ straight) (NN rod)))))))))
    (VP (VBZ is)
      (VP (VBN unclosed)))
    (. .)))
```

Figure 2: Example for class 2 on Syntactic level.

In Figure 2, subject-verb pair '*(opening-2 is-13)*' can be concluded from dependency parsing '*nsubjpass(unclosed-14, opening-2)*' and '*auxpass(unclosed-14, is-13)*'. In the same way, the other pair '*(which-6 is-7)*' is obtained, too. However, the problem is that '*which-6*' is not the true subject capable to evaluate if the subject-verb is in agreement, the true one should be ‘*hook-5*’. But no links between ‘*which-6*’and ‘*hook-5*’ is served in the parsing above in Figure 2. As regards to this kind outcome as ‘*(which-6 is-7)*’, we re-recognize the subject-verb after reverting the *wh-* word back to the most possible sentence element that *wh-* word points to.

3.2 Sentence Simplification

Sentence simplification is an interesting point in this paper. Grefenstette (1998) applies shallow parsing and simplification rules to the problem of telegraphic text reduction, with as goal the development of an audio scanner for the blind or for people using their sight for other tasks like driving. Another related application area is the shorting of text to fit the screen of mobile devices (Corston-Oliver, 2001; Euler 2002).

We employ the sentence simplification as a pre-processing operation by deleting some kinds of adjective, adverb, modified noun and some kind prepositional phrase, so that the sentence becomes more simple with the trunk elements, such as the subject, the verbs and the object, left. By analyzing the training data, a positive simplification categories set is picked out and shown as follows:

Table 1: Categories to simplify a sentence.

#	Original	Delete	#	Original	Delete
1	RB$_1$ CC RB$_2$ JJ	RB$_1$ CC RB$_2$	11	DT JJ CC VBG NN*	JJ CC VBG
2	RB$_1$ JJ|RB$_2$|MD	RB$_1$	12	!VB JJ$_1$ NN*|JJ$_2$	JJ$_1$
3	DT NN$_1$ CC NN$_2$ NN*	CC NN$_2$	13	, JJ ,	JJ,
4	!IN&&!TO NN|CD NN (!%)	NN|CD	14	JJ$_1$ VBG NN*|JJ$_2$	JJ$_1$ VBG
5	NN VBP|VBZ|VBG VBP|VBZ	NN	15	DT VBG$_1$ CC VBG$_2$ NN*|JJ	VBG$_1$ CC VBG$_2$
6	NN VBP|VBZ JJ IN	NN	16	DT VBD VBG NN	VBD VBG
7	DT NN* CC NN VBN NN*	CC NN$_1$ VBN	17	DT VBG|VBN JJ|NN*	VBG|VBN
8	DT NN$_1$ VBG$_1$ CC VBG$_2$ NN$_2$	NN$_1$ VBG$_1$ CC VBG$_2$	18	only VB*(is|are|am|was|were)	“Only” TO “there”
9	DT NN$_1$ VBG|VBN NN$_2$	NN$_1$ VBG|VBN	19	NN* PP (not with VB* in)	PP (not with VB* in)
10	JJ$_1$ CC JJ$_2$ JJ$_3$|NN	JJ$_1$ CC JJ$_2$	#		

In Table 1, the ‘Original’ POS sequence can be regarded as triggering environment, ‘Delete’ points to the sequence that should be deleted. And the signal ‘!’ is not a punctuation, but as a logic operator. ‘NN*’ means NN or NNS. In addition, the simplification operation of ‘JJ’, ‘NN’, ‘VB*’, ‘RB’ or their POS sequence is done based on POS, while the operation of ‘PP’ chunk is done based on Phrase Structure Parsing.

The best target of sentence simplification are sentences that are totally correctly tagged (POS) and parsed (Phrase Structure Parsing). For those incorrectly done, inappropriate simplification outcome appear. But since incorrectly done, no matter whether the simplification operation is correct, it will not decline the system performance. So, we make each sentence in the corpus simplified.

3.3 wh-type Word Reverting

The *wh-* words, such as “which”, “who”, “what” and “that”, usually exist in a sentence as the subject, and if the sentence is a subordinate clause, a more detailed sentence subject should be found. In order to obtain a much exacter subject, we do a reverting operation to the *wh-* word.

Firstly, retrieve the most possible subject element in the sentence that *wh-* word may point to. Secondly, replace *wh-* word with the subject element and extract the subordinate clauses to be independent, so that a complicate and long sentence becomes several relative simple ones. Then, discover the subject-verb pairs of all the new generated sentences by making dependency grammar analysis. Terminally, we combine the subject-verb pairs back into the outcome of the original sentence.

The algorithm for reverting is as follows:

```
Input: Phrase syntactic parsing file;
Output: The element that wh- word most possibly points to, only NP is
        considered here;
int Distan(WDT, $NP_i$)              // the distance between WDT and $NP_i$;
                              // WDT is the Part Of Speech of wh- word;
  begin
  // weight of each branch w = 1;
  // Value_Distance(node1,node2) = w × the number of branches connect
      node1 and node2;
  Definition: int distance = 0;
    if node P as the nearest and common ancestor of WDT and $NP_i$;
        distance = Value_Distance(P,WDT) + Value_Distance(P,$NP_i$);
        return distance;
    else
        return +∞;
    end if
end

string Revert()
 Definition: int dis;
             int DIS;  // the distance between the wh- word and the NP;
             string SUBJ;   // the most possible NP wh- word points to;
             SUBJ = Null, DIS = +∞;
  begin
  for each $NP_i$ before the wh- word in Parsed-Tree // $NP_i$ must before the
                                                // wh- word in the sentence;
      dis = Distan(WDT, $NP_i$); // calculate the distance of $NP_i$ and WDT;
      If dis < DIS            // search the nearest $NP_i$;
          DIS = dis;
          SUBJ = $NP_i$;
      else
          continue;
      end if
  end for
  return SUBJ;
end
```

4 Experiments

How the subject and the predicate verb link up with each other in a sentence is rather flexible, especially for the science and technology literature sentences, such as patent corpus, which are too long and with too many modifiers in. This makes the subject-verb agreement evaluation more difficult. In this paper, we utilize the patent corpus.

4.1 Development Data

It is mainly used for learning the sentence simplification categories. By analyzing the tagging and parsing outcome of the sentences given, we choose categories that positively function to simplifying a sentence to be a set, as in Table 1. Totally, 600 manually proofread English patent sentences are used to develop the categories set.

4.2 Evaluation Data

For the evaluation, we experiment on 1000 English patent sentences translated by non-native speakers. In order to make a general comparison, the corpus is separated into four different parts as follows:

Table 2: Analysis of evaluation corpus.

corpus	Short sentences		Long sentences	
Number(sen.)	332 sen.		668 sen.	
Percentage(%)	33.2%		66.8%	
	subject-verb pairs agreed	subject-verb pairs disagreed	subject-verb pairs agreed	subject-verb pairs disagreed
Number(sen.)	172 sen.	160 sen.	328 sen.	340 sen.
Percentage(%)	17.2%	16%	32.8%	34%

Note: Long sentence: if length of the sentence > 40 words;
Short sentence: if length of the sentence < 40 words.

In order to compute the precision of the system outcome, we annotate the correct subject-verb pairs and their agreement of the 1000 sentences manually as the reference. E.g., for the sentence in Figure 2, it is ‘*opening-2 is-13 1|hook-5 is-7 1|*’, where ‘*1*|’ means the subject-verb is in agreement, ‘*0*|’ means disagreement in contrast.

4.3 Evaluation Metric

According to the common three evaluation guidelines, the following statistics are computed as the criterion to evaluate the performance of the system:

Precision The proportion of the system subject-verb pairs which are correct. Calculated as follows:

$$P = \frac{N}{M} \times 100\% \tag{1}$$

Note: N is the number of the correct subject-verb pairs in system outcome. *M* is the total number of the subject-verb pairs in system outcome.

Recall Out of all the subject-verb pairs in the reference, the proportion that appear in the system outcome. Calculated as follows:

$$R = \frac{N}{T} \times 100\% \tag{2}$$

Note: T is the total number of the subject-verb pairs in the reference.

F-Score Which is a combination of *P* and *R*, and is a more general evaluation score. The formula is as follows:

$$F = \frac{P \times R \times (\beta^2 + 1)}{R + \beta^2 \times P} \times 100\% \tag{3}$$

Note: β is an important weight parameter between *P* and *R*, it is regarded as 1 in this paper, i.e. *P* and *R* share the same weight.

4.4 Experiment Setting

The experiment is implemented as following steps:

Step 1 Pre-Processing Tokenize the patent corpus in §4.2.

Step 2 Phrase Syntactic Parsing e.g. *The opening of existing hook which is hanged on a straight rod is unclosed, and the under frame, the tension spring, the swing arm and the tensile force constant device are all equipped in the protecting cover.* *(1)*

Parsed with Stanford-parser:

(ROOT (S (S (NP (NP (DT The) (NN opening)) (PP (IN of) (NP (VBG existing) (NN hook) (SBAR (WHNP (WDT which)) (S (VP (VBZ is) (VP (VBN hanged) (PP (IN on) (NP (DT a) (JJ straight) (NN rod)))))))))) (VP (VBZ is) (VP (VBN unclosed)))) (, ,) (CC and) (S (NP (DT the) (ADJP (JJ under) (NP (NP (NN frame)) (, ,) (NP (DT the) (NN tension) (NN spring)) (, ,) (NP (DT the) (NN swing) (NN arm)) (CC and) (NP (DT the) (JJ tensile) (NN force)))) (JJ constant) (NN device)) (VP (VBP are) (RB all) (ADJP (VBN equipped) (PP (IN in) (NP (DT the) (JJ protecting) (NN cover)))))) (. .))) *(2)*

Step 3 Sentence Simplification. Simplify the sentences by deleting some elements, such as some kind *JJ* or *NN* or *RB* or *PP* chunk that listed in Table 1. As is simplified, *(2)* becomes into *(3)*:

The opening of existing hook which is hanged on a rod is unclosed, and the frame, the spring, the arm and the force device are equipped in the cover. *(3)*

Step 4 Do Dependency Parsing to sentence *(3)*, the subjects and their predicate verbs are linked up, and subject-verb pairs:

'*opening-2 is-12 |which-6 is-7 |frame-17 are-28 |spring-20 are-28 |arm-23 are-28 |device-27 are-28* |' *(4)*

can be recognized.

Step 5 Revert the *wh-* subject For the pairs such as '*which-6 is-7* |' in which *wh*-type subject is recognized, the sentence will be rechecked by reverting the *wh-* word back into the word or chunk (usually as *NP* chunk before the *wh-* word) that the *wh-* word most possibly points to. Once the *wh-* word is reverted, retrieve the subordinate clauses to be independent. **Go to step 2.**

For the outcome of *(4)*, '*which-6*' is replaced as '*hook-5*', and the original sentence becomes:

The opening of existing hook is unclosed, and the frame , the spring , the arm and the force device are equipped in the cover. *(5)*

and *Existing hook is hanged on a rod.* *(6)*

Since both are rechecked, combine the subject-verb pairs of *(5)* and *(6)* to be:

opening-2 is-12 |hook-5 is-7 |frame-17 are-28 |spring-20 are-28 |arm-23 are-28 |device-27 are-28 | *(7)*

Step 6 Terminal outcome Evaluate if the subject-verb pairs are in agreement according to their POS (Part Of Speech). According to *(2)*, the POS of *(7)* is:

NN VBZ |NN VBZ | NN VBP |NN VBP | NN VBP | NN VBP |

So, the agreement outcome is:

opening-2 is-12 1|hook-5 is-7 1|frame-17 are-28 0|spring-20 are-28 0|arm-23 are-28 0|device-27 are-28 0| *(8)*

Note: '0|' stands for disagreement; '1|' stands for agreement.

In addition, four different subjects in *(8)* share the same verb '*are-28*', it is a plural case. So, their agreement labels should be modified to '*1*|'. Then, the terminal result comes to be:

opening-2 is-12 1|hook-5 is-7 1|frame-17 are-28 1|spring-20 are-28 1|arm-23 are-28 1|device-27 are-28 1| *(9)*

5 The Experimental Results and Analysis

Table 3 compares the outcomes of different phases of the subject-verb discovery: the first one is merely based on dependency grammar; sentence simplification is added to be the second one; and the third one adds *wh*-type word reverting operation to the second. Outcome of the first is present as the baseline.

Table 3: The outcome of the subject-verb discovery.

	Dep.				SSIM+Dep.				SSIM+ Dep.+WH-.			
	Short sentences		Long sentences		Short sentences		Long sentences		Short sentences		Long sentences	
Subject-verb agreed(Y/N)	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N
R(%)	96.89	96.33	91.05	91.12	96.89	96.33	93.68	90.89	96.89	97.91	93.82	91.13
P(%)	93.96	92.93	89.29	85.68	94.92	94.85	92.35	86.14	94.92	96.89	92.48	86.66
F(%)	**95.41**	**94.60**	**90.16**	**88.32**	**95.90**	**95.58**	**93.01**	**88.45**	**95.90**	**97.40**	**93.14**	**88.84**
R_{total}(%)	92.16				93.07				93.38			
P_{total}(%)	88.53				90.16				90.63			
F_{total}(%)	**90.31**				**91.59**				**91.98**			

Note: **Dep.** Means Dependency Parsing; **SSIM** means Sentence Simplification; **WH-** is the operation of rechecking of *wh-* type subject.

The comparison of the subject-verb agreement evaluation on the pairs that correctly recognized is as follows in Table 4:

Table 4: Precision of agreement evaluation on the subject-verb pairs that correctly recognized.

	Dep.				SSIM+ Dep.				SSIM+ Dep.+WH-.			
	Short sentences		Long sentences		Short sentences		Long sentences		Short sentences		Long sentences	
Subject-verb agreed(Y/N)	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N
P(%)	99.47	97.27	97.53	97.88	99.47	97.81	97.04	98.28	99.47	97.86	97.04	98.41
P_{total}(%)	**97.86**				**97.88**				**97.93**			

In Table 3, the subject-verb discovery outcomes of the three methods are presented, including the Precision(P), Recall(R) and F-score(F) on each subset of the corpus, as well as the total F-score on the whole corpus. In Table 4, it is the precision of the subject-verb agreement evaluation based on the subject-verb pairs that have been recognized correctly in Table 3.

By comparison, the figures show that both the SSIM and WH- operations function positively that the final F_{total} of the recognition improves 1.67%. And from the percentage it improves step by step, SSIM is shown to get a more remarkable F_{total}. This is because every sentence can be simplified while not all of them contain a *wh-* subordinate clause, actually there are only 269 *wh-* words in the corpus. Moreover, the categories for SSIM must be selected carefully, or else it may result in negative effect. But WH- is always positive, since it only aims at the incorrect subject-verb recognition. However, maybe there could be more appropriate categories for SSIM or more perfect method for WH-, on that the system will perform better.

As to the subject-verb pairs that is discovered correctly, for the reason of the precision of Part Of Speech tagging, the agreement evaluation is impossible to be whole correct. The Precision(P) on the subsets of the corpus and the whole corpus are as Table 4.

6 Conclusion

Subject-verb agreement is a complicated and difficult problem in Machine Translation Evaluation, it is involved with complicated grammar, long dependency relationship, and subordinate clause factors, and so on. Especially for the science and technology literature sentences, such as patent corpus, which are too long or with too many modifiers in, it gets worse.

We have proposed a hybrid method for subject-verb agreement evaluation on dependency grammars with the processing of phrase syntactic parsing and sentence simplification for subject-verb discovery. It is completely automatically done, and the results show its efficiency.

By the way, the categories we use for sentence simplification and *wh-* type subject reverting operation may be not much appropriate, the better categories are made, the better the system performs.

References

Atwell, E. S. 1987. How to detect grammatical errors in a text without parsing it. *Proceeding of the 3rd EACL*. 38-45.

Bigert, J. and O. Knutsson. 2002. Robust error detection: A Hybrid Approach Combining unsupervised error detection and linguistic knowledge. *Proceeding of Robust Method in Analysis of Natural Language Data.* 10-19.

Bender, E., D. Flickinger, S. Oepen, A. Walsh, and T. Baldwin. 2004. Arboretum: Using a Precision Grammar for Grammar Checking in CALL. *Proc. In-STIL/ICALL Symposium on Computer Assisted Learning.*

Santorini, B. 1990. Part Of Speech Tagging Guidelines for the Penn Treebank Project (3rd Version, 2nd Printing). *http://bulba.sdsu.edu/jeanette/thesis/PennTags.html.*

Chodorow, M. and C. Leacock. 2000. An Unsupervised Method for detecting Grammatical Errors. *Proceeding of NAACL'00.* 140-147.

de Marneffe, M.-C. and C.D. Manning. 2008. *Stanford typed dependencies manual-[EB].*

Heidorn, G. 2000. Intelligent Writing Assistance. *Handbook of Natural Language Processing.* Obert Dale, Hermann Moisi and Harold Somers (ed.). Marcel Dekker, Inc.

Izumi, E., K. Uchimoto, T. Saiga, T. Supnithi, and H. Isahara. 2003. Automatic Error Detection in the Japanese Learner's English Spoken Data. *Companion Volume to Proc. ACL*. Sapporo, Japan.

Lee, J. and S. Seneff. 2008. Correcting Misuse of Verb Forms. *22nd International Conference on Computational Linguistics.*

Lee, J. and S. Seneff. 2006. Automatic Grammar Correction for Second-Language Learners. *Proc. Interspeech*. Pittsburgh, PA.

Michaud, L., K. McCoy, and C. Pennington. 2000. An Intelligent Tutoring System for Deaf Learners of Written English. *Proc. 4th International ACM Conference on Assistive Technologies*.

Tesniere, L. 1959. *Elements de Syntaxe Structurale*. Paris: Klincksieck.