IITB System for CoNLL 2013 Shared Task: A Hybrid Approach to Grammatical Error Correction

Anoop Kunchukuttan **Ritesh Shah** **Pushpak Bhattacharyya**
Department of Computer Science and Engineering, IIT Bombay
{anoopk,ritesh,pb}@cse.iitb.ac.in

Abstract

We describe our grammar correction system for the CoNLL-2013 shared task. Our system corrects three of the five error types specified for the shared task - noun-number, determiner and subject-verb agreement errors. For noun-number and determiner correction, we apply a classification approach using rich lexical and syntactic features. For subject-verb agreement correction, we propose a new rule-based system which utilizes dependency parse information and a set of conditional rules to ensure agreement of the verb group with its subject. Our system obtained an F-score of 11.03 on the official test set using the M^2 evaluation method (the official evaluation method).

1 Introduction

Grammatical Error Correction (GEC) is an interesting and challenging problem and the existing methods that attempt to solve this problem take recourse to deep linguistic and statistical analysis. In general, GEC may partly assist in solving natural language processing (NLP) tasks like Machine Translation, Natural Language Generation *etc*. However, a more evident application of GEC is in building automated grammar checkers thereby benefiting non-native speakers of a language. The CoNLL-2013 shared task (Ng et al., 2013) looks at improving the current approaches for GEC and for inviting novel perspectives towards solving the same. The shared task makes the NUCLE corpus (Dahlmeier et al., 2013) available in the public domain and participants have been asked to correct grammatical errors belonging to the following categories: noun-number, determiner, subject-verb agreement (SVA), verb form and preposition. The key challenges are handling interaction between different error groups and handling potential mistakes made by off-the-shelf NLP components run on erroneous text.

For the shared task, we have addressed the following problems: noun-number, determiner and subject-verb agreement correction. For noun-number and determiner correction, we use a classification based approach to predict corrections - which is a widely used approach (Knight and Chander, 1994; Rozovskaya and Roth, 2010). For subject-verb agreement correction, we propose a new rule-based approach which applies a set of conditional rules to correct the verb group to ensure its agreement with its subject. Our system obtained a score of 11.03 on the official test set using the M^2 method. Our SVA correction system performs very well with a F-score of 28.45 on the official test set.

Section 2 outlines our approach to solving the grammar correction problem. Sections 3, 4 and 5 describe the details of the noun-number, determiner and SVA correction components of our system. Section 6 explains our experimental setup. Section 7 discusses the results of the experiments and Section 8 concludes the report.

2 Problem Formulation

In this work, we focus on correction of three error categories related to nouns: noun-number, determiner and subject-verb agreement. The number of the noun, the choice of determiner and verb's agreement in number with the subject are clearly inter-related. Therefore, a coordinated approach is necessary to correct these errors. If these problems are solved independently of each other, wrong corrections may be generated. The following are some examples:

Erroneous sentence
A good workmen does not blame his tools
Good corrections
A good workman does not blame his tools
Good workmen do not blame his tools

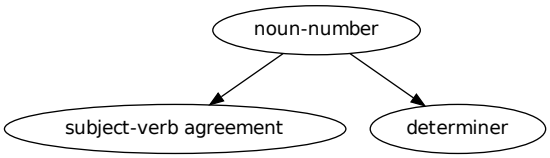


Figure 1: Dependencies between the noun-number, determiner and subject-verb agreement errors

Bad corrections
A good workman do not blame his tools
Good workman does not blame his tools

The choice of noun-number is determined by the discourse and meaning of the text. The choice of determiner is partly determined by the noun-number, whereas the verb's agreement depends completely on the number of its subject. Figure 1 shows the proposed dependencies between the number of a noun, its determiner and number agreement with the verb for which the noun is the subject. Assuming these dependencies, we first correct the noun-number. The corrections to the determiner and the verb's agreement with the subject are done taking into consideration the corrected noun. The noun-number and determiner are corrected using a classification based approach, whereas the SVA errors are corrected using a rule-based system; these are described in the following sections.

3 Noun Number Correction

The major factors which determine the number of the noun are: (i) the intended meaning of the text, (ii) reference to the noun earlier in the discourse, and (iii) stylistic considerations. Grammatical knowledge is insufficient for determining the noun-number, which requires a higher level of natural language processing. For instance, consider the following examples:
(1) I bought all the recommended books. These books are costly.
(2) Books are the best friends of man.

In Example (1), the choice of plural noun in the second sentence is determined by a reference to the entity in the previous sentence. Example (2) is a general statement about a class of entities, where the noun is generally a plural. Such phenomena make noun-number correction a difficult task. As information at semantic and discourse levels is difficult to encode, we explored lexical and syntactic

Tokens, POS and chunk tags in ± 2 word-window around the noun
Is the noun capitalized ?
Is the noun an acronym ?
Is the noun a named entity?
Is the noun a mass noun, *pluralia tantum*?
Does the noun group have an article/ demonstrative/quantifier?
What article/demonstrative/quantifier does the noun phrase have ?
Are there words indicating plurality in the context of the noun?
The first two words of the sentence and their POS tags
The number of the verb for which this noun is the subject
Grammatical Number of majority of nouns in noun phrase conjunction

Table 1: Feature set for noun-number correction

information to obtain cues about the number of the noun. The following is a summary of the cues we have investigated:
Noun properties: Is the noun a mass noun, a *pluralia tantum*, a named entity or an acronym?
Lexical context: The presence of a plurality indicating word in the context of the noun (e.g. the ancient *scriptures* **such as** the Vedas, Upanishads, etc.)
Syntactic constraints:

- Nouns linked by a conjunction agree with each other (e.g. *The pens, pencils and books*).
- Presence/value of the determiner in the noun group. However, this is only a secondary cue, since it is not possible to determine if it is the determiner or the noun-number that is incorrect (e.g. *A books*).
- Agreement with the verb of which the noun is the subject. This is also a secondary feature.

Given that we are dealing with erroneous text, these cues could themselves be wrong. The problem of noun-number correction is one of making a prediction based on multiple cues in the face of such uncertainty. We model the problem as a binary classification problem, the task being to predict if the observed noun-number of every noun in the text needs correction (labels: *requires_correction/no_correction*). Alterna-

tively, we could formulate the problem as a *singular/plural* number prediction problem, which would not require annotated learner corpora text. However, we prefer the former approach since we can learn corrections from learner corpora text (as opposed to native speaker text) and use knowledge of the observed number for prediction. Use of observed values has been shown to be beneficial for grammar correction (Rozovskaya and Roth, 2010; Dahlmeier and Ng, 2011).

If the model predicts *requires_correction*, then the observed number is toggled to obtain the corrected noun-number. In order to bias the system towards improved precision, we apply the correction only if classifier's confidence score for the *requires_correction* prediction exceeds its score for the *no_correction* prediction by at least a threshold value. This threshold value is determined empirically. The feature set designed for the classifier is shown in Table 1.

4 Determiner Correction

Determiners in English consist of articles, demonstratives and quantifiers. The choice of determiners, especially articles, depends on many factors including lexical, syntactic, semantic and discourse phenomena (Han et al., 2006). Therefore, the correct usage of determiners is difficult to master for second language learners, who may (i) insert a determiner where it is not required, (ii) omit a required determiner, or (iii) use the wrong determiner. We pose the determiner correction problem as a classification problem, which is a well explored method (Han et al., 2006; Dahlmeier and Ng, 2011). Every noun group is a training instance, with the determiner as the class label. Absence of a determiner is indicated by a special class label *NO_DET*. However, since the number of determiners is large, a single multi-class classifier will result in ambiguity. This ambiguity can be reduced by utilizing of the fact that a particular observed determiner is replaced by one of a small subset of all possible determiners (which we call its *confusion set*). For instance, the confusion set for *a* is {*a, an, the, NO_DET*}. It is unlikely that *a* is replaced by any other determiner like *this, that*, *etc*. Rozovskaya and Roth (2010) have used this method for training preposition correction systems, which we adopt for training a determiner correction system. For each observed determiner, we build a classifier whose prediction is limited to the confusion set of the observed determiner. The confusion sets were obtained from the training corpus. The feature set is almost the same as the one for noun-number correction. The only difference is that context window features (token, POS and chunk tags) are taken around the determiner instead of the noun.

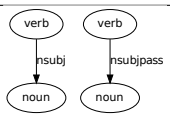
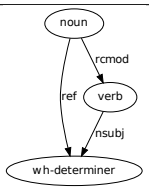
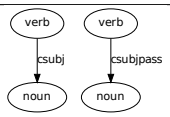
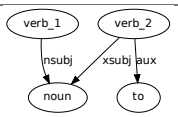
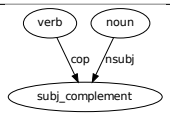
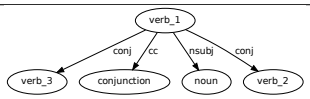
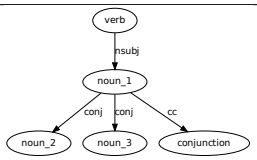
	Description	Path
1	Direct subject	
2	Path through Wh-determiner	
3	Clausal subject	
4	External subject	
5	Path through copula	
6	Subject in a different clause	
7	Multiple subjects	

Table 2: Some rules from the *singularize_verb_group* rule-set

5 Subject-Verb Agreement

The task in subject-verb agreement correction is to correct the verb group components so that it agrees with its subject. The correction could be made either to the verb inflection (*He run* → *He runs*) or to the auxiliary verbs in the verb group (*He are running* → *He is running*). We assume that noun-number and verb form errors (tense, aspect, modality) do not exist or have already been corrected. We built a rule-based system for performing SVA correction, whose major components are (i) a system for detecting the subject of a verb, and

(ii) a set of conditional rules to correct the verb group.

We use a POS tagger, constituency parser and dependency parser for obtaining linguistic information (noun-number, noun/verb groups, dependency paths) required for SVA correction. Our assumption is that these NLP tools are reasonably robust and do a good analysis when presented with erroneous text. We have used the Stanford suite of tools for the shared task and found that it makes few mistakes on the NUCLE corpus text.

The following is our proposed algorithm for SVA correction:

1. Identify noun groups in a sentence and the information associated with each noun group: (i) number of the head noun of the noun group, (ii) associated noun groups, if the noun group is part of a noun phrase conjunction, and (iii) head and modifier in each noun group pair related by the *if* relation.

2. Identify the verb groups in a sentence.

3. For every verb group, identify its subject as described in Section 5.1.

4. If the verb group does not agree in number with its subject, correct each verb group by applying the conditional rules described in Section 5.2.

5.1 Identifying the subject of the verb

We utilize dependency relations (uncollapsed) obtained from the Stanford dependency parser to identify the subject of a verb. From analysis of dependency graphs of sentences in the NUCLE corpus, we identified different types of dependency paths between a verb and its subject, which are shown in Table 2. Given these possible dependency path types, we identify the subject of a verb using the following procedure:

- First, check if the subject can be reached using a direct dependency path (paths (1), (2), (3) and (4))

- If a direct relation is not found, then look for a subject via path (5)

- If the subject has not been found in the previous step, then look for a subject via path (6)

A verb can have multiple subjects, which can be identified via dependency path (7).

Rule	Condition	Action
1	$\exists w \in vg, \text{pos\_tag}(w) = \mathbf{MD}$	Do nothing
2	$\exists w \in vg, \text{pos\_tag}(w) = \mathbf{TO}$	Do nothing
3	$\text{subject}(vg) \neq \mathbf{I}$	Replace *are* by *is*
4	$\text{subject}(vg) = \mathbf{I}$	Replace *are* by *am*
5	$\text{do, does} \notin vg \wedge \text{subject}(vg) \neq \mathbf{I}$	Replace *have* by *has*
6	$\text{do, does} \notin vg \wedge \text{subject}(vg) = \mathbf{I}$	Replace *has* by *have*

Table 3: Some rules from the *singularize_verb_group* rule-set
w is a word, vg is a verb group, POS tags are from the Penn tagset

5.2 Correcting the verb group

For correcting the verb group, we have two sets of conditional rules (*singularize_verb_group* and *pluralize_verb_group*). The *singularize_verb_group* rule-set is applied if the subject is singular, whereas the *pluralize_verb_group* rule-set is applied if the subject is plural or if there are multiple subjects (path (7) in Table 2). For verbs which have subjects related via dependency paths (3) and (4) no correction is done.

The conditional rules utilize POS tags and lemmas in the verb group to check if the verb group needs to be corrected and appropriate rules are applied for each condition. Some rules in the *singularize_verb_group* rule-set are shown in Table 3. The rules for the *pluralize_verb_group* rule-set are analogous.

6 Experimental Setup

Our training data came from the NUCLE corpus provided for the shared task. The corpus was split into three parts: training set (55151 sentences), threshold tuning set (1000 sentences) and development test set (1000 sentences). In addition, evaluation was done on the official test set (1381 sentences). Maximum Entropy classifiers were trained for noun-number and determiner correction systems. In the training set, the number of instances with no corrections far exceeds the number of instances with corrections. Therefore, a balanced training set was created by including all the instances with corrections and sampling α instances with no corrections from the training set. By trial and error, α was determined to be 10000 for the noun-number and determiner correction systems. The confidence score threshold which maximizes the F-score was calibrated on the tuning set. We determined $threshold = 0$

Task	Development test set			Official test set		
	P	R	F-1	P	R	F-1
Noun Number	31.43	40	35.2	28.47	9.84	14.66
Determiner	35.59	17.5	23.46	21.43	1.3	2.46
SVA	16.67	23.42	19.78	29.57	27.42	28.45
Integrated	**29.59**	**17.24**	**21.79**	**28.18**	**4.99**	**11.03**

Table 4: M^2 scores for IIT Bombay correction system: component-wise and integrated

for the noun-number and the determiner correction systems.

The following tools were used in the development of the system for the shared task: (i) NLTK (MaxEntClassifier, Wordnet lemmatizer), (ii) Stanford tools - POS Tagger, Parser and NER and Python interface to the Stanford NER, (iii) Lingua::EN::Inflect module for noun and verb pluralization, and (iv) Wiktionary list of mass nouns, *pluralia tantum*.

7 Results and Discussion

Table 4 shows the results on the test set (development and official) for each component of the correction system and the integrated system. The evaluation was done using the M^2 method (Dahlmeier and Ng, 2012). This involves computing F1 measure between a set of proposed system edits and a set of human-annotated gold-standard edits. However, evaluation is complicated by the fact that there may be multiple edits which generate the same correction. The following example illustrates this behaviour:

Source: I ate mango
Hypothesis: I ate **a** mango

The system edit is $\epsilon \rightarrow a$, whereas the gold standard edit is *mango→a mango*. Though both the edits result in the same corrected sentence, they do not match. The M^2 algorithm resolves this problem by providing an efficient method to detect the sequence of phrase-level edits between a source sentence and a system hypothesis that achieves the highest overlap with the gold-standard annotation.

It is clear that the low recall of the noun-number and determiner correction components have resulted in a low overall score for the system. This underscores the difficulty of the two problems. The feature sets seem to have been unable to capture the patterns determining the noun-number and determiner. Consider a few examples, where the evidence for correction look strong:

1. products such as RFID tracking **system** have become real

2. With the installing of the **surveillances** for every corner of Singapore

A cursory inspection of the corpus indicates that in the absence of a determiner (example (1)), the noun tends to be plural. This pattern has not been captured by the correction system. The coverage of the *Wiktionary* mass noun and *pluralia tantum* dictionaries is low, hence this feature has not had the desired impact (example(2)).

The SVA correction component has a reasonably good precision and recall - performing best amongst all the correction components. Since most errors affecting agreement (noun-number, verb form, etc.) were not corrected, the SVA agreement component could not correct the agreement errors. If these errors had been corrected, the accuracy of the standalone SVA correction component would have been higher than that indicated by the official score. To verify this, we manually analyzed the output from the SVA correction component and found that 58% of the missed corrections and 43% of the erroneous corrections would not have occurred if some of the other related errors had been fixed. If it is assumed that all these errors are corrected, the effective accuracy of SVA correction increases substantially as shown in Table 5. A few errors in the gold standard for SVA agreement were also considered for computing the effective scores. The standalone SVA correction module therefore has a good accuracy.

A major reason for SVA errors ($\sim$18%) is wrong output from NLP modules like the POS tagger, chunker and parser. The following are a few examples:

- The verb group is incorrectly identified if there is an adverb between the main and auxiliary verbs.

  *It [do **not only** restrict] their freedom in all*

SVA Score	Development test set			Official test set		
	P	R	F-1	P	R	F-1
Official	16.67	23.42	19.78	29.57	27.42	28.45
Effective	51.02	55.55	53.18	65.32	66.94	66.12

Table 5: M^2 scores (original and modified) for SVA correction

aspects , but also causes leakage of personal information .

- Two adjacent verb groups are not distinguished as separate chunks by the chunker when the second verb group is non-finite involving an infinitive.

  *The police arrested all of them before they [starts **to** harm] the poor victim.*

- The dependency parser makes errors in identifying the subject of a verb. The noun *problems* is not identified as the subject of *is* by the dependency parser.

  *Although rising of life expectancies is an challenge to the entire human nation , the detailed **problems** each country that will encounter **is** different.*

Some phenomena have not been handled by our rules. Our system does not handle the case where the subject is a gerund phrase. Consider the example,

Collecting coupons from individuals are the first step.

The verb-number should be singular when a gerund phrase is the subject. In the absence of rules to handle this case, *coupons* is identified as the subject of *are* by the dependency parser and consequently, no correction is done.

Our rules do not handle interrogative sentences and interrogative pronouns. Hence the following sentence is not corrected,

People do not know who are tracking them.

Table 6 provides an analysis of the error type distribution for SVA errors on the official test set.

8 Conclusion

In this paper, we presented a hybrid grammatical correction system which incorporates both machine learning and rule-based components. We proposed a new rule-based method for subject-verb agreement correction. As future work, we plan to explore richer features for noun-number and determiner errors.

Error types	% distribution
Noun-number errors	58.02 %
Wrong tagging, chunking, parsing	18.52 %
Wrong gold annotations	7.40%
Rules not designed	6.1%
Others	9.88 %

Table 6: Causes for missed SVA corrections and their distribution in the official test set

References

Daniel Dahlmeier and Hwee Tou Ng. 2011. Grammatical error correction with alternating structure optimization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*.

Daniel Dahlmeier and Hwee Tou Ng. 2012. Better evaluation for grammatical error correction. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics*.

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 2013. Building a large annotated corpus of learner english: The NUS Corpus of Learner English. In *To appear in Proceedings of the 8th Workshop on Innovative Use of NLP for Building Educational Applications*.

Na-Rae Han, Martin Chodorow, and Claudia Leacock. 2006. Detecting errors in english article usage by non-native speakers. *Natural Language Engineering*.

Kevin Knight and Ishwar Chander. 1994. Automated postediting of documents. In *AAAI*.

Hwee Tou Ng, Siew Mei Wu, Yuanbin Wu, Christian Hadiwinoto, and Joel Tetreault. 2013. The CoNLL-2013 Shared Task on Grammatical Error Correction. In *To appear in Proceedings of the Seventeenth Conference on Computational Natural Language Learning*.

Alla Rozovskaya and Dan Roth. 2010. Generating confusion sets for context-sensitive error correction. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*.