Using Contextual Speller Techniques and Language Modeling for ESL Error Correction

Michael Gamon*, Jianfeng Gao*, Chris Brockett*, Alexandre Klementiev⁺, William B. Dolan*, Dmitriy Belenko*, Lucy Vanderwende*

*Microsoft Research
One Microsoft Way
Redmond, WA 98052
{mgamon,jfgao,chrisbkt,billdol,
dmitryb,lucyv}@microsoft.com

⁺Dept. of Computer Science
University of Illinois
Urbana, IL 61801
klementi@uiuc.edu

Abstract

We present a modular system for detection and correction of errors made by non-native (English as a Second Language = ESL) writers. We focus on two error types: the incorrect use of determiners and the choice of prepositions. We use a decision-tree approach inspired by contextual spelling systems for detection and correction suggestions, and a large language model trained on the Gigaword corpus to provide additional information to filter out spurious suggestions. We show how this system performs on a corpus of non-native English text and discuss strategies for future enhancements.

1 Introduction

English is today the de facto lingua franca for commerce around the globe. It has been estimated that about 750M people use English as a second language, as opposed to 375M native English speakers (Crystal 1997), while as much as 74% of writing in English is done by non-native speakers. However, the errors typically targeted by commercial proofing tools represent only a subset of errors that a non-native speaker might make. For example, while many non-native speakers may encounter difficulty choosing among prepositions, this is typically not a significant problem for native speakers and hence remains unaddressed in proofing tools such as the grammar checker in Microsoft Word (Heidorn 2000). Plainly there is an opening here for automated proofing tools that are better geared to the non-native users.

One challenge that automated proofing tools face is that writing errors often present a semantic dimension that renders it difficult if not impossible to provide a single correct suggestion. The choice of definite versus indefinite determiner—a common error type among writers with a Japanese, Chinese or Korean language background owing to the lack of overt markers for definiteness and indefiniteness—is highly dependent on larger textual context and world knowledge. It seems desirable, then, that proofing tools targeting such errors be able to offer a range of plausible suggestions, enhanced by presenting real-world examples that are intended to inform a user's selection of the most appropriate wording in the context[1].

2 Targeted Error Types

Our system currently targets eight different error types:

1. Preposition presence and choice:
   In the other hand, ... (On the other hand ...)
2. Definite and indefinite determiner presence and choice:
   I am teacher... (am a teacher)
3. Gerund/infinitive confusion:
   I am interesting in this book. (interested in)
4. Auxiliary verb presence and choice:
   My teacher does is a good teacher (my teacher is...)

[1] Liu et al. 2000 take a similar approach, retrieving example sentences from a large corpus.

5. Over-regularized verb inflection:
   I writed a letter (wrote)
6. Adjective/noun confusion:
   This is a China book (Chinese book)
7. Word order (adjective sequences and nominal compounds):
   I am a student of university (university student)
8. Noun pluralization:
   They have many knowledges (much knowledge)

In this paper we will focus on the two most prominent and difficult errors: choice of determiner and prepositions. Empirical justification for targeting these errors comes from inspection of several corpora of non-native writing. In the NICT Japanese Learners of English (JLE) corpus (Izumi et al. 2004), 26.6% of all errors are determiner related, and about 10% are preposition related, making these two error types the dominant ones in the corpus. Although the JLE corpus is based on transcripts of spoken language, we have no reason to believe that the situation in written English is substantially different. The Chinese Learners of English Corpus (CLEC, Gui and Yang 2003) has a coarser and somewhat inconsistent error tagging scheme that makes it harder to isolate the two errors, but of the non-orthographic errors, more than 10% are determiner and number related. Roughly 2% of errors in the corpus are tagged as preposition-related, but other preposition errors are subsumed under the “collocation error” category which makes up about 5% of errors.

3 Related Work

Models for determiner and preposition selection have mostly been investigated in the context of sentence realization and machine translation (Knight and Chander 1994, Gamon et al. 2002, Bond 2005, Suzuki and Toutanova 2006, Toutanova and Suzuki 2007). Such approaches typically rely on the fact that preposition or determiner choice is made in otherwise native-like sentences. Turner and Charniak (2007), for example, utilize a language model based on a statistical parser for Penn Tree Bank data. Similarly, De Felice and Pulman (2007) utilize a set of sophisticated syntactic and semantic analysis features to predict 5 common English prepositions. Obviously, this is impractical in a setting where noisy non-native text is subjected to proofing. Meanwhile, work on automated error detection on non-native text focuses primarily on detection of errors, rather than on the more difficult task of supplying viable corrections (e.g., Chodorow and Leacock, 2000). More recently, Han et al. (2004, 2006) use a maximum entropy classifier to propose article corrections in TESOL essays, while Izumi et al. (2003) and Chodorow et al. (2007) present techniques of automatic preposition choice modeling. These more recent efforts, nevertheless, do not attempt to integrate their methods into a more general proofing application designed to assist non-native speakers when writing English. Finally, Yi et al. (2008) designed a system that uses web counts to determine correct article usage for a given sentence, targeting ESL users.

4 System Description

Our system consists of three major components:

1. Suggestion Provider (SP)
2. Language Model (LM)
3. Example Provider (EP)

The Suggestion Provider contains modules for each error type discussed in section 2. Sentences are tokenized and part-of-speech tagged before they are presented to these modules. Each module determines parts of the sentence that may contain an error of a specific type and one or more possible corrections. Four of the eight error-specific modules mentioned in section 2 employ machine learned (classification) techniques, the other four are based on heuristics. Gerund/infinitive confusion and auxiliary presence/choice each use a single classifier. Preposition and determiner modules each use two classifiers, one to determine whether a preposition/article should be present, and one for the choice of preposition/article.

All suggestions from the Suggestion Provider are collected and passed through the Language Model. As a first step, a suggested correction has to have a higher language model score than the original sentence in order to be a candidate for being surfaced to the user. A second set of heuristic thresholds is based on a linear combination of class probability as assigned by the classifier and language model score.

The Example Provider queries the web for exemplary sentences that contain the suggested correction. The user can choose to consult this information to make an informed decision about the correction.

4.1 Suggestion Provider Modules for Determiners and Prepositions

The SP modules for determiner and preposition choice are machine learned components. Ideally, one would train such modules on large data sets of annotated errors and corrected counterparts. Such a data set, however, is not currently available. As a substitute, we are using native English text for training, currently we train on the full text of the English Encarta encyclopedia (560k sentences) and a random set of 1M sentences from a Reuters news data set. The strategy behind these modules is similar to a contextual speller as described, for example, in (Golding and Roth 1999). For each potential insertion point of a determiner or preposition we extract context features within a window of six tokens to the right and to the left. For each token within the window we extract its relative position, the token string, and its part-of-speech tag. Potential insertion sites are determined heuristically from the sequence of POS tags. Based on these features, we train a classifier for preposition choice and determiner choice. Currently we train decision tree classifiers with the WinMine toolkit (Chickering 2002). We also experimented with linear SVMs, but decision trees performed better overall and training and parameter optimization were considerably more efficient. Before training the classifiers, we perform feature ablation by imposing a count cutoff of 10, and by limiting the number of features to the top 75K features in terms of log likelihood ratio (Dunning 1993).

We train two separate classifiers for both determiners and preposition:

- decision whether or not a determiner/preposition should be present (*presence/absence* or ***pa classifier****)*
- decision which determiner/preposition is the most likely choice, given that a determiner/preposition is present (*choice* or ***ch classifier***)

In the case of determiners, class values for the ch classifier are *a/an* and *the*. Preposition choice (equivalent to the “confusion set” of a contextual speller) is limited to a set of 13 prepositions that figure prominently in the errors observed in the JLE corpus: *about, as, at, by, for, from, in, like, of, on, since, to, with, than, "other"* (for prepositions not in the list).

The decision tree classifiers produce probability distributions over class values at their leaf nodes. For a given leaf node, the most likely preposition/determiner is chosen as a suggestion. If there are other class values with probabilities above heuristically determined thresholds[2], those are also included in the list of possible suggestions. Consider the following example of an article-related error:

I am teacher from Korea.

As explained above, the suggestion provider module for article errors consists of two classifiers, one for presence/absence of an article, the other for article choice. The string above is first tokenized and then part-of-speech tagged:

0/I/PRP 1/am/VBP 2/teacher/NN 3/from/IN 4/Korea/NNP 5/./.

Based on the sequence of POS tags and capitalization of the nouns, a heuristic determines that there is one potential noun phrase that could contain an article: *teacher*. For this possible article position, the article presence/absence classifier determines the probability of the presence of an article, based on a feature vector of pos tags and surrounding lexical items:

p(article + *teacher*) = 0.54

Given that the probability of an article in this position is higher than the probability of not having an article, the second classifier is consulted to provide the most likely choice of article:

p(*the*) = 0.04
p(*a/an*) = 0.96

Given this probability distribution, a correction suggestion *I am teacher from Korea -> I am **a** teacher from Korea* is generated and passed on to evaluation by the language model component.

4.2 The Language Model

The language model is a 5-gram model trained on the English Gigaword corpus (LDC2005T12). In order to preserve (singleton) context information as much as possible, we used interpolated Kneser-Ney smoothing (Kneser and Ney 1995) without count cutoff. With a 120K-word vocabulary, the trained language model contains 54 million bigrams, 338 million trigrams, 801 million 4-grams

[2] Again, we are working on learning these thresholds empirically from data.

and 12 billion 5-grams. In the example from the previous section, the two alternative strings of the original user input and the suggested correction are scored by the language model:

I am teacher from Korea. score = 0.19
*I am **a** teacher from Korea.* score = 0.60

The score for the suggested correction is significantly higher than the score for the original, so the suggested correction is provided to the user.

4.3 The Example Provider

In many cases, the SP will produce several alternative suggestions, from which the user may be able to pick the appropriate correction reliably. In other cases, however, it may not be clear which suggestion is most appropriate. In this event, the user can choose to activate the Example Provider (EP) which will then perform a web search to retrieve relevant example sentences illustrating the suggested correction. For each suggestion, we create an exact string query including a small window of context to the left and to the right of the suggested correction. The query is issued to a search engine, and the retrieved results are separated into sentences. Those sentences that contain the string query are added to a list of example candidates. The candidates are then ranked by two initially implemented criteria: Sentence length (shorter examples are preferred in order to reduce cognitive load) and context overlap (sentences that contain additional words from the user input are preferred). We have not yet performed a user study to evaluate the usefulness of the examples provided by the system. Some examples of usage that we retrieve are given below with the query string in boldface:

Original: *I am teacher from Korea.*
Suggestion: *I am a teacher from Korea.*
All top 3 examples: ***I am a teacher**.*

Original: *So Smokers have to see doctor more often than non-smokers.*
Suggestion: *So Smokers have to see a doctor more often than non-smokers.*
Top 3 examples:

1. *Do people going through withdrawal have to **see a doctor**?*
2. *Usually, a couple should wait to **see a doctor** until after they've tried to get pregnant for a year.*
3. *If you have had congestion for over a week, you should **see a doctor**.*

Original: *I want to travel Disneyland in March.*
Suggestion: *I want to travel **to** Disneyland in March.*
Top 3 examples:

1. *Timothy's wish was **to travel to Disneyland** in California.*
2. *Should you **travel to Disneyland** in California or to Disney World in Florida?*
3. *The tourists who **travel to Disneyland** in California can either choose to stay in Disney resorts or in the hotel for Disneyland vacations.*

5 Evaluation

We perform two different types of evaluation on our system. Automatic evaluation is performed on native text, under the assumption that the native text does not contain any errors of the type targeted by our system. For example, the original choice of preposition made in the native text would serve as supervision for the evaluation of the preposition module. Human evaluation is performed on non-native text, with a human rater assessing each suggestion provided by the system.

5.1 Individual SP Modules

For evaluation, we split the original training data discussed in section 4.1 into training and test sets (70%/30%). We then retrained the classifiers on this reduced training set and applied them to the held-out test set. Since there are two models, one for preposition/determiner presence and absence (*pa*), and one for preposition/determiner choice (*ch*), we report combined accuracy numbers of the two classifiers. *Votes(a)* stands for the counts of votes for class value = *absence* from *pa*, *votes(p)* stands for counts of votes for *presence* from *pa*. *Acc(pa)* is the accuracy of the *pa* classifier, *acc(ch)* the accuracy of the choice classifier. Combined accuracy is defined as in Equation 1.

$$\frac{acc(pa) * votes(a) + acc(ch) * acc(pa) * votes(p)}{total\ cases}$$

Equation 1: Combined accuracy of the presence/absence and choice models

The total number of cases in the test set is 1,578,342 for article correction and 1,828,438 for preposition correction.

5.1.1 Determiner choice

Accuracy of the determiner pa and ch models and their combination is shown in Table 1.

Model	pa	ch	combined
Accuracy	89.61%	85.97%	86.07%

Table 1: Accuracy of the determiner pa, ch, and combined models.

The baseline is 69.9% (choosing the most frequent class label *none*). The overall accuracy of this module is state-of-the-art compared with results reported in the literature (Knight and Chander 1994, Minnen et al. 2000, Lee 2004, Turner and Charniak 2007). Turner and Charniak 2007 obtained the best reported accuracy to date of 86.74%, using a Charniak language model (Charniak 2001) based on a full statistical parser on the Penn Tree Bank. These numbers are, of course, not directly comparable, given the different corpora. On the other hand, the distribution of determiners is similar in the PTB (as reported in Minnen et al. 2000) and in our data (Table 2).

	PTB	Reuters/Encarta mix
no determiner	70.0%	69.9%
the	20.6%	22.2%
a/an	9.4%	7.8%

Table 2: distribution of determiners in the Penn Tree Bank and in our Reuters/Encarta data.

Precision and recall numbers for both models on our test set are shown in Table 3 and Table 4.

Article pa classifier	precision	recall
presence	84.99%	79.54%
absence	91.43%	93.95%

Table 3: precision and recall of the article pa classifier.

Article ch classifier	precision	Recall
the	88.73%	92.81%
a/an	76.55%	66.58%

Table 4: precision and recall of the article ch classifier.

5.1.2 Preposition choice

The preposition choice model and the combined model achieve lower accuracy than the corresponding determiner models, a result that can be expected given the larger choice of candidates and hardness of the task. Accuracy numbers are presented in Table 5.

Model	pa	ch	combined
Accuracy	91.06%%	62.32%	86.07%

Table 5:Accuracy of the preposition pa, ch, and combined models.

The baseline in this task is 28.94% (using no preposition). Precision and recall numbers are shown in Table 6 and Table 7. From Table 7 it is evident that prepositions show a wide range of predictability. Prepositions such as *than* and *about* show high recall and precision, due to the lexical and morphosyntactic regularities that govern their distribution. At the low end, the semantically more independent prepositions *since* and *at* show much lower precision and recall numbers.

Preposition pa classifier	precision	recall
presence	90.82%	87.20%
absence	91.22%	93.78%

Table 6: Precision and recall of the preposition pa classifier.

Preposition ch classifier	precision	recall
other	53.75%	54.41%
in	55.93%	62.93%
for	56.18%	38.76%
of	68.09%	85.85%
on	46.94%	24.47%
to	79.54%	51.72%
with	64.86%	25.00%
at	50.00%	29.67%
by	42.86%	60.46%
as	76.78%	64.18%
from	81.13%	39.09%
since	50.00%	10.00%
about	93.88%	69.70%
than	95.24%	90.91%

Table 7: Precision and recall of the preposition ch classifier.

Chodorow et al. (2007) present numbers on an independently developed system for detection of preposition error in non-native English. Their approach is similar to ours in that they use a classifier with contextual feature vectors. The major differences between the two systems are the additional use of a language model in our system and, from a usability perspective, in the example provider module we added to the correction process. Since both systems are evaluated on different data sets[3], however, the numbers are not directly comparable.

5.2 Language model Impact

The language model gives us an additional piece of information to make a decision as to whether a correction is indeed valid. Initially, we used the language model as a simple filter: any correction that received a lower language model score than the original was filtered out. As a first approximation, this was an effective step: it reduced the number of preposition corrections by 66.8% and the determiner corrections by 50.7%, and increased precision dramatically. The language model alone, however, does not provide sufficient evidence: if we produce a full set of preposition suggestions for each potential preposition location and rank these suggestions by LM score alone, we only achieve 58.36% accuracy on Reuters data.

Given that we have multiple pieces of information for a correction candidate, namely the class probability assigned by the classifier and the language model score, it is more effective to combine these into a single score and impose a tunable threshold on the score to maximize precision. Currently, this threshold is manually set by analyzing the flags in a development set.

5.3 Human Evaluation

A complete human evaluation of our system would have to include a thorough user study and would need to assess a variety of criteria, from the accuracy of individual error detection and corrections to the general helpfulness of real web-based example sentences. For a first human evaluation of our system prototype, we decided to simply address the question of accuracy on the determiner and preposition choice tasks on a sample of non-native text.

For this purpose we ran the system over a random sample of sentences from the CLEC corpus (8k for the preposition evaluation and 6k for the determiner evaluation). An independent judge annotated each flag produced by the system as belonging to one of the following categories:

- (1) the correction is valid and fixes the problem
- (2) the error is correctly identified, but the suggested correction does not fix it
- (3) the original and the rewrite are both equally good
- (4) the error is at or near the suggested correction, but it is a different kind of error (not having to do with prepositions/determiners)
- (5) There is a spelling error at or near the correction
- (6) the correction is wrong, the original is correct

Table 8 shows the results of this human assessment for articles and prepositions.

	Articles (6k sentences)		**Prepositions (8k sentences)**	
	count	**ratio**	**count**	**ratio**
(1) correction is valid	240	55%	165	46%
(2) error identified, suggestion does not fix it	10	2%	17	5%
(3) original and suggestion equally good	17	4%	38	10%
(4) misdiagnosis	65	15%	46	13%
(5) spelling error near correction	37	8%	20	6%
(6) original correct	70	16%	76	21%

Table 8: Article and preposition correction accuracy on CLEC data.

The distribution of corrections across deletion, insertion and substitution operations is illustrated in Table 9. The most common article correction is insertion of a missing article. For prepositions, substitution is the most common correction, again an expected result given that the presence of a

[3] Chodorow et al. (2007) evaluate their system on proprietary student essays from non-native students, where they achieve 77.8% precision at 30.4% recall for the preposition substitution task.

preposition is easier to determine for a non-native speaker than the actual choice of the correct preposition.

	deletion	insertion	substitution
Articles	8%	79%	13%
Prepositions	15%	10%	76%

Table 9: Ratio of deletion, insertion and substitution operations.

6 Conclusion and Future Work

Helping a non-native writer of English with the correct choice of prepositions and definite/indefinite determiners is a difficult challenge. By combining contextual speller based methods with language model scoring and providing web-based examples, we can leverage the combination of evidence from multiple sources.

The human evaluation numbers presented in the previous section are encouraging. Article and preposition errors present the greatest difficulty for many learners as well as machines, but can nevertheless be corrected even in extremely noisy text with reasonable accuracy. Providing contextually appropriate real-life examples alongside with the suggested correction will, we believe, help the non-native user reach a more informed decision than just presenting a correction without additional evidence and information.

The greatest challenge we are facing is the reduction of “false flags”, i.e. flags where both error detection and suggested correction are incorrect. Such flags—especially for a non-native speaker—can be confusing, despite the fact that the impact is mitigated by the set of examples which may clarify the picture somewhat and help the users determine that they are dealing with an inappropriate correction. In the current system we use a set of carefully crafted heuristic thresholds that are geared towards minimizing false flags on a development set, based on detailed error analysis. As with all manually imposed thresholding, this is both a laborious and brittle process where each retraining of a model requires a re-tuning of the heuristics. We are currently investigating a learned ranker that combines information from language model and classifiers, using web counts as a supervision signal.

7 Acknowledgements

We thank Claudia Leacock (Butler Hill Group) for her meticulous analysis of errors and human evaluation of the system output, as well as for much invaluable feedback and discussion.

References

Bond, Francis. 2005. *Translating the Untranslatable: A Solution to the Problem of Generating English Determiners.* CSLI Publications.

Charniak, Eugene. 2001. Immediate-head parsing for language models. In *Proceedingsof the 39th Annual Meeting of the Association for Computational Linguistics*, pp 116-123.

Chickering, David Maxwell. 2002. The WinMine Toolkit. Microsoft Technical Report 2002-103.

Chodorow, Martin, Joel R. Tetreault and Na-Rae Han. 2007. Detection of Grammatical Errors Involving Prepositions. In *Proceedings of the 4th ACL-SIGSEM Workshop on Prepositions*, pp 25-30.

Crystal, David. 1997. *Global English.* Cambridge University Press.

Rachele De Felice and Stephen G Pulman. 2007. *Automatically acquiring models of preposition use.* Proceedings of the ACL-07 Workshop on Prepositions.

Dunning, Ted. 1993. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 19:61-74.

Gamon, Michael, Eric Ringger, and Simon Corston-Oliver. 2002. Amalgam: A machine-learned generation module. Microsoft Technical Report, MSR-TR-2002-57.

Golding, Andrew R. and Dan Roth. 1999. A Winnow Based Approach to Context-Sensitive Spelling Correction. *Machine Learning*, pp. 107-130.

Gui, Shicun and Huizhong Yang (eds.). 2003. *Zhongguo Xuexizhe Yingyu Yuliaohu*. (*Chinese Learner English Corpus*). Shanghai Waiyu Jiaoyu Chubanshe..

Han, Na-Rae., Chodorow, Martin and Claudia Leacock. 2004. Detecting errors in English article usage with a maximum entropy classifier trained on a large, diverse corpus. *Proceedings of the 4th international conference on language resources and evaluation*, Lisbon, Portugal.

Han, Na-Rae. Chodorow, Martin., and Claudia Leacock. (2006). Detecting errors in English article usage by non-native speakers. *Natural Language Engineering, 12*(2), 115-129.

Heidorn, George. 2000. Intelligent Writing Assistance. In Robert Dale, Herman Moisl, and Harold Somers (eds.). *Handbook of Natural Language Processing.* Marcel Dekker. pp 181 -207.

Izumi, Emi, Kiyotaka Uchimoto and Hitoshi Isahara. 2004. The NICT JLE Corpus: Exploiting the Language Learner's Speech Database for Research and Education. *International Journal of the Computer, the Internet and Management* 12:2, pp 119 -125.

Kneser, Reinhard. and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1. 1995. pp. 181–184.

Knight, Kevin and Ishwar Chander. 1994. Automatic Postediting of Documents. *Proceedings of the American Association of Artificial Intelligence,* pp 779-784.

Lee, John. 2004. Automatic Article Restoration. *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 31-36.

Liu, Ting, Mingh Zhou, JianfengGao, Endong Xun, and Changning Huan. 2000. PENS: A Machine-Aided English Writing System for Chinese Users. *Proceedings of ACL 2000*, pp 529-536.

Minnen, Guido, Francis Bond and Ann Copestake. 2000. Memory-Based Learning for Article Generation. *Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic* Workshop, pp 43-48.

Suzuki, Hisami and Kristina Toutanova. 2006. Learning to Predict Case Markers in Japanese. *Proceedings of* COLING-ACL, pp. 1049-1056.

Toutanova, Kristina and Hisami Suzuki. 2007 Generating Case Markers in Machine Translation. *Proceedings of NAACL-HLT.*

Turner, Jenine and Eugene Charniak. 2007. Language Modeling for Determiner Selection. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pp 177-180.

Yi, Xing, Jianfeng Gao and William B. Dolan. 2008. Web-Based English Proofing System for English as a Second Language Users. *To be presented at IJCNLP 2008.*