Using Mostly Native Data to Correct Errors in Learners' Writing: A Meta-Classifier Approach

Michael Gamon
Microsoft Research
One Microsoft Way
Redmond, WA 98052
mgamon@microsoft.com

Abstract

We present results from a range of experiments on article and preposition error correction for non-native speakers of English. We first compare a language model and error-specific classifiers (all trained on large English corpora) with respect to their performance in error detection and correction. We then combine the language model and the classifiers in a meta-classification approach by combining evidence from the classifiers and the language model as input features to the meta-classifier. The meta-classifier in turn is trained on error-annotated learner data, optimizing the error detection and correction performance on this domain. The meta-classification approach results in substantial gains over the classifier-only and language-model-only scenario. Since the meta-classifier requires error-annotated data for training, we investigate how much training data is needed to improve results over the baseline of not using a meta-classifier. All evaluations are conducted on a large error-annotated corpus of learner English.

1 Introduction

Research on the automatic correction of grammatical errors has undergone a renaissance in the past decade. This is, at least in part, based on the recognition that non-native speakers of English now outnumber native speakers by 2:1 in some estimates, so any tool in this domain could be of tremendous value. While earlier work in both native and non-native error correction was focused on the construction of grammars and analysis systems to detect and correct specific errors (see Heift and Schulze, 2005 for a detailed overview), more recent approaches have been based on data-driven methods.

The majority of the data-driven methods use a classification technique to determine whether a word is used appropriately in its context, continuing the tradition established for contextual spelling correction by Golding (1995) and Golding and Roth (1996). The words investigated are typically articles and prepositions. They have two distinct advantages as the subject matter for investigation: They are a closed class and they comprise a substantial proportion of learners' errors. The investigation of preposition corrections can even be narrowed further: amongst the more than 150 English prepositions, the usage of the ten most frequent prepositions accounts for 82% of preposition errors in the 20 million word *Cambridge University Press Learners' Corpus*. Learning correct article use is most difficult for native speakers of an L1 that does not overtly mark definiteness and indefiniteness as English does. Prepositions, on the other hand, pose difficulties for language learners from all L1 backgrounds (Dalgish, 1995; Bitchener et al., 2005).

Contextual classification methods represent the context of a preposition or article as a feature vector gleaned from a window of a few words around the preposition/article. Different systems typically vary along three dimensions: choice of features, choice of classifier, and choice of training data. Features range from words and morphological information (Knight and Chander, 1994) to the inclusion of part-of-speech tags (Minnen et al., 2000; Han et al., 2004, 2006; Chodorow et al., 2007; Gamon et al., 2008, 2009; Izumi et al., 2003, 2004; Tetrault and Chodorow, 2008) to features based on linguistic analysis and on WordNet (Lee, 2004; DeFelice and Pulman, 2007, 2008). Knight and Chander (1994) and Gamon et al. (2008) used decision tree classifiers but, in general, maximum entropy classifiers have become the classification

algorithm of choice. Training data are normally drawn from sizeable corpora of native English text (*British National Corpus* for DeFelice and Pulman (2007, 2008), *Wall Street Journal* in Knight and Chander (1994), a mix of *Reuters* and *Encarta* in Gamon et al. (2008, 2009). In order to partially address the problem of domain mismatch between learners' writing and the news-heavy data sets often used in data-driven NLP applications, Han et al. (2004, 2006) use 31.5 million words from the MetaMetrics corpus, a diverse corpus of fiction, non-fiction and textbooks categorized by reading level.

In addition to the classification approach to error detection, there is a line of research - going back to at least Atwell (1987) - that uses language models. The idea here is to detect errors in areas where the language model score is suspiciously low. Atwell (1987) uses a part-of-speech tag language model to detect errors, Chodorow and Leacock (2000) use mutual information and chi square statistics to identify unlikely function word and part-of-speech tag sequences, Turner and Charniak (2007) employ a language model based on a generative statistical parser, and Stehouwer and van Zaanen (2009) investigate a diverse set of language models with different backoff strategies to determine which choice, from a set of confusable words, is most likely in a given context. Gamon et al. (2008, 2009) use a combination of error-specific classifiers and a large generic language model with hand-tuned heuristics for combining their scores to maximize precision. Finally, Yi et al. (2008) and Hermet et al. (2008) use n-gram counts from the web as a language model approximation to identify likely errors and correction candidates.

2 Our Approach

We combine evidence from the two kinds of data-driven models that have been used for error detection and correction (error-specific classifiers and a language model) through a meta-classifier. We use the term *primary models* for both the initial error-specific classifiers and a large generic language model. The *meta-classifier* takes the output of the primary models (language model scores and class probabilities) as input. Using a meta-classifier for ensemble learning has been proven effective for many machine learning problems (see e.g. Dietterich 1997), especially when the combined models are sufficiently different to make distinct kinds of errors. The meta-classification approach also has an advantage in terms of data requirements: Our primary models are trained on large sets of widely available well-formed English text. The meta-classifier, in contrast, is trained on a smaller set of error-annotated learner data. This allows us to address the problem of domain mismatch: We can leverage large well-formed data sets that are substantially different from real-life learner language for the primary models, and then fine-tune the output to learner English using a much smaller set of expensive and hard-to-come-by annotated learner writing.

For the purpose of this paper, we restrict ourselves to article and preposition errors. The questions we address are:

1. How effective is the meta-classification approach compared to either a classifier or a language model alone?
2. How much error-annotated data are sufficient to produce positive results above the baseline of using either a language model or a classifier alone?

Our evaluation is conducted on a large data set of error-annotated learner data.

3 Experimental Design

3.1 Primary Models

Our error-specific primary models are maximum entropy classifiers (Rathnaparki 1997) for articles and for prepositions. Features include contextual features from a window of six tokens to the right and left, such as lexical features (word), part-of-speech tags, and a handful of "custom features", for example lexical head of governing VP or governed NP (as determined by part-of-speech-tag based heuristics). For both articles and prepositions, we employ two classifiers: the first determines the probability that a preposition/article is present in a given context (*presence classifier*), the second classifier determines the probability that a specific article or preposition is chosen (*choice classifier*). A training event for the presence classifier is any noun phrase boundary that is a potential location for a preposition or article. Whether a location is an NP boundary and a potential site for an article/preposition is determined by a simple heuristic based on part-of-speech tags.

The candidates for article choice are *the* and *a/an*, and the choice for prepositions is limited to twelve very frequent prepositions (*in, at, on, for, since, with, to, by, about, from, of, as*) which account for 86.2 % of preposition errors in our learner data. At prediction time, the presence and choice classifiers produce a list of potential changes in preposition/article usage for the given context. Since the application of our system consists of suggesting corrections to a user, we do not consider identity operations where the suggested word choice equals the actual word choice. For a potential preposition/article location where there is no preposition/article, each of the candidates is considered for an insertion operation. For a potential location that contains a preposition/article, the possible operations include deletion of the existing token or substitution with another preposition/article from the candidate set. Training data for the classifiers is a mix of primarily well-formed data sources: There are about 2.5 million sentences, distributed roughly equally across *Reuters* newswire, *Encarta* encyclopedia, UN proceedings, *Europarl* and web-scraped general domain data[1]. From the total set of candidate operations (substitutions, insertions, and deletions) that each combination of presence and choice classifier produces for prepositions, we consider only the top three highest-scoring operations[2].

Our language model is trained on the *Gigaword* corpus (Linguistic Data Consortium, 2003) and utilizes 7-grams with absolute discount smoothing (Gao, Goodman, and Miao, 2001; Nguyen, Gao, and Mahajan, 2007). Each suggested revision from the preposition/article classifiers (top three for prepositions, all revisions from the article classifiers) are scored by the language model: for each revision, the language model score of the original and the suggested rewrite is recorded, as is the language model entropy (defined as the language model probability of the sentence, normalized by sentence length).

[1] We are not able to train the error-specific classifiers on a larger data set like the one we use for the language model. Note that the 2.5 million sentences used in the classifier training already produce 16.5 million training vectors.

[2] This increases runtime performance because fewer calls need to be made to the language model which resides on a server. In addition, we noticed that overall precision is increased by not considering the less likely suggestions by the classifier.

3.2 Meta-Classifier

For the meta-classifier we chose to use a decision tree, trained with the WinMine toolkit (Chickering 2002). The motivation for this choice is that decision trees are well-suited for continuously valued features and for non-linear decision surfaces. An obvious alternative would be to use a support vector machine with non-linear kernels, a route that we have not explored yet. The feature set for the meta-classifier consists of the following scores from the primary models, including some arithmetic combinations of scores:

- Ratio and delta of Log LM score of the original word choice and the suggested revision (2 features)
- Ratio and delta of the LM entropy for original and suggested revision (2 features).
- Products of the above ratios/deltas and classifier choice/presence probabilities
- Type of operation: deletion, insertion, substitution (3 features)
- P(presence) (1 feature)
- For each preposition/article choice: P(choice): 13 features for prepositions (12 prepositions and *other* for a preposition not in that set), 2 for articles
- Original token: *none* (for insertion) or the original preposition/article (13 features for prepositions, 2 for articles)
- Suggested token: *none* (for deletion) or the suggested preposition/article (13 features for prepositions, 2 for articles)

The total number of features is 63 for prepositions and 36 for articles.

The meta-classifier is trained by collecting suggested corrections from the primary models on the error annotated data. The error-annotation provides the binary class label, i.e. whether the suggested revision is correct or incorrect. If the suggested revision matches an annotated correction, it counts as correct, if it does not match it counts as incorrect. To give an example, the top three preposition operations for the position before *this test* in the sentence *I rely to this test* are:

Change_to_on
Delete_to
Change_to_of

The class label in this example is "suggestion correct", assuming that the change of preposition is

annotated in the data. The operation *Change_to_on* in this example has the following feature values for the basic classifier and LM scores:

classifier P(choice): 0.755
classifier P(presence): 0.826
LM logP(original): -17.373
LM logP(rewrite): -14.184

An example of a path through the decision tree meta-classifier for prepositions is:

LMLogDelta is Not < -8.59 and
LMLogDelta is Not < -3.7 and
ProductRewriteLogRatioConf is Not < -0.00115 and
LMLogDelta is Not < -1.58 and
ProductOrigEntropyRatioChoiceConf is Not < -0.00443 and
choice_prob is Not < 0.206 and
Original_of is 0 and
choice_prob is Not < 0.329 and
to_prob is < 0.108 and
Suggested_on is 1 and
Original_in is 0 and
choice_prob is Not < 0.497 and
choice_prob is Not < 0.647 and
presence_prob is Not < 0.553

The leaf node at the end of this path has a 0.21 probability of changing "*to*" to "*on*" being a correct rewrite suggestion.

The features selected by the decision trees range across all of the features discussed above. For both the article and preposition meta-classifiers, the ranking of features by importance (as measured by how close to the root the decision tree uses the feature) follows the order in which features are listed.

3.3 Data

In contrast to the training data for the primary models, the meta-classifier is trained on error-annotated data from the *Cambridge University Press Learners' Corpus* (CLC). The version of CLC that we have licensed currently contains a total of 20 million words from learner English essays written as part of one of *Cambridge's English Language Proficiency Tests* (ESOL) – at all proficiency levels. The essays are annotated for error type, erroneous span and suggested correction.

We first perform a random split of the essays into 70% training, 20% test and 10% for parameter tuning. Next, we create error-specific training, tuning and test sets by performing a number of clean-up steps on the data. First, we correct all errors that were flagged as being spelling errors, since we presume that the user will perform a spelling check on the data before proceeding to grammatical proofing. Spelling errors that were flagged as morphology errors were left alone. By the same token, we corrected confused words that are covered by MS Word. We then revised British English spelling to American English spelling conventions. In addition, we eliminated all annotations for non-pertinent errors (i.e., non-preposition/article errors, or errors that do not involve any of the targeted prepositions), but we maintained the original (erroneous) text for these. This makes our task harder since we will have to learn how to make predictions in text containing multiple errors, but it also is a more realistic scenario given real learner writing. Finally, we eliminated sentences containing nested errors and immediately adjacent errors when they involve pertinent (preposition/article) errors. For example, an annotated error "*take a picture*" with the correction "*take pictures*" is annotated as two consecutive errors: "delete *a*" and "rewrite *picture* as *pictures*". Since the error involves operations on both the article and the noun, which our article correction module is not designed to cover, we eliminated the sentence from the data. (This last step eliminated 31% of the sentences annotated with preposition errors and 29% or the sentences annotated with article errors.) Sentences that were flagged for a replacement error but contained no replacement were also eliminated from the data.

The final training, tuning and test set sizes are as follows (note that for prepositions we had to reduce the size of the training set by an additional 20% in order to avoid memory limitations of our decision tree tools).

Prepositions:

train: 584,485 sentences, 68,806 prep errors
tuning: 105,166 sentences, 9918 prep errors
test: 208,724 sentences, 19,706 prep errors

Articles:

train: 737,091 sentences, 58,356 article errors
tuning: 106,052 sentences, 8341 article errors
test: 210,577 sentences, 16,742 article errors

This mix is strongly biased towards "correct" usage. After all, there are many more correct uses of articles and prepositions in the CLC data than incorrect ones. Again, this is likely to make our task harder, but more realistic, since both at train-

ing and test time we are working with the error distribution that is observed in learner data.

3.4 Evaluation

To evaluate, we run our meta-classifier system on the preposition and article test sets described in above and calculate precision and recall. Precision and recall for the overall system are controlled by thresholding the meta-classifier class probability. As a point of comparison, we also evaluate the performance of the primary models (the error-specific classifier and the language model) in isolation. Precision and recall for the error-specific classifier is controlled by thresholding class probability. To control the precision-recall tradeoff for the language model, we calculate the difference between the log probabilities of the original user input and the suggested correction. We then vary that difference across all observed values in small increments, which affects precision and recall: the higher the difference, the fewer instances we find, but the higher the reliability of these instances is.

This evaluation differs from many of the evaluations reported in the error detection/correction literature in several respects. First, the test set is a broad random sample across all proficiency levels in the CLC data. Second, it is far larger than any sets that have been so far to report results of preposition/article correction on learner data. Finally, we are only considering cases in which the system suggests a correction. In other words, we do not count as correct instances where the system's prediction matches a correct preposition/article.

This evaluation scheme, however, ignores one aspect of a real user scenario. Of all the suggested changes that are counted as wrong in our evaluation because they do not match an annotated error, some may in fact be innocuous or even helpful for a real user. Such a situation can arise for a variety of reasons: In some cases, there are legitimate alternative ways to correct an error. In other cases, the classifier has identified the location of an error although that error is of a different kind (which can be beneficial because it causes the user to make a correction - see Leacock et al., 2009). Gamon et al. (2009), for example manually evaluate preposition suggestions as belonging to one of three categories: (a) properly correcting an existing error, (b) offering a suggestion that neither improves nor degrades the user sentence, (c) offering a suggestion that would degrade the user input. Obviously, (c) is a more serious error than (b). Similarly, Tetrault and Chodorow (2008) annotate their test set with preposition choices that are valid alternatives. We do not have similar information in the CLC data, but we can perform a manual analysis of a random subset of test data to estimate an "upper bound" for our precision/recall curve. Our annotator manually categorized each suggested correction into one of seven categories.

Details of the distribution of suggested corrections into the seven categories are shown in Table 1.

Category	preps.	articles
Corrects a CLC error	32.87%	33.34%
Corrects an error that was not annotated as being that error type in CLC	11.67%	12.16%
Corrects a CLC error, but uses an alternative correction	3.62%	2.26%
Original and suggested correction are equally good	9.60%	11.30%
Error correctly detected, but the correction is wrong	8.73%	5.03%
Identifies an error site, but the actual error is not a preposition error	19.17%	12.64%
Introduces an error	14.65%	23.26%

Table 1: Manual analysis of suggested corrections on CLC data.

This analysis involves costly manual evaluation, so we only performed it at one point of the precision/recall curve (our current runtime system setting). The sample size was 6,000 sentences for prepositions and 5981 sentences for articles (half of the sentences were flagged as containing at least one article/preposition error while the other half were not). On this manual evaluation, we achieve 32.87% precision if we count all flags that do not perfectly match a CLC annotation as a false positive. Only counting the last category (introduction of an error) as a false positive, precision is at 85.34%. Similarly, for articles, the manual estimation arrives at 76.74% precision, where pure CLC annotation matching gives us 33.34%.

4 Results

Figure 1 and Figure 2 show the evaluation results of the meta-classifier for prepositions and articles, compared to the performance of the error-specific classifier and language model alone. For both prepositions and articles, the first notable observation is that the language model outperforms the classifier by a large margin. This came as a surprise to us, given the recent prevalence of classification approaches in this area of research and the fact that our classifiers produce state-of-the art performance when compared to other systems, on well-formed data. Second, the combination of scores from the classifier and language model through a meta-classifier clearly outperforms either one of them in isolation. This result, again, is consistent across prepositions and articles.

We had previously used a hand-tuned score combination instead of a meta-classifier. We also established that this heuristic performs worse than the language model for prepositions, and just about at the same level as the language model for articles. Note, though, that the manual tuning was performed to optimize performance against a different data set (the *Chinese Learners of English Corpus*: CLEC), so the latter point is not really comparable and hence is not included in the charts.

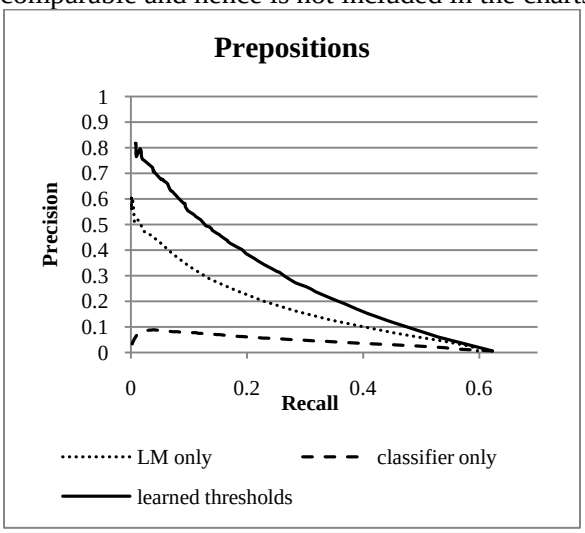

Figure 1: Precision and recall for prepositions.

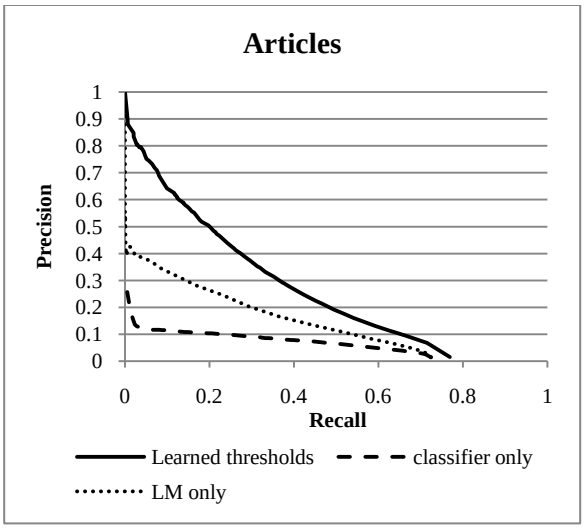

Figure 2: Precision and recall for articles.

We now turn to the question of the required amount of annotated training data for the meta-classifier. CLC is commercially available, but it is obvious that for many researchers such a corpus will be too expensive and they will have to create or license their own error-annotated corpus. Thus the question of whether one could use less annotated data to train a meta-classifier and still achieve reasonable results becomes important. Figure 3 and Figure 4 show results obtained by using decreasing amounts of training data. The dotted line shows the language model baseline. Any result below the language model performance shows that the training data is insufficient to warrant the use of a meta-classifier. In these experiments there is a clear difference between prepositions and articles. We can reduce the amount of training data for prepositions to 10% of the original data and still outperform the language model baseline. 10% of the data corresponds to 6,800 annotated preposition errors and 58,400 sentences. When we reduce the training data to 1% of the original amount (680 annotated errors, 5,800 sentences) we clearly see degraded results compared to the language model. With articles, the system is much less data-hungry. Reducing the training data to 1% (580 annotated errors, 7,400 sentences) still outperforms the language model alone. This result can most likely be explained by the different complexity of the preposition and article tasks. Article operations include only six distinct operations: deletion of *the*, deletion of *a/an*, insertion of *the*, insertion of *a/an*, change of *the* to *a/an*, and change of *a/an* to *the*. For the twelve prepositions that we work with, the

total number of insertions, deletions and substitutions that require sufficient training events and might need different score combinations is 168, making the problem much harder.

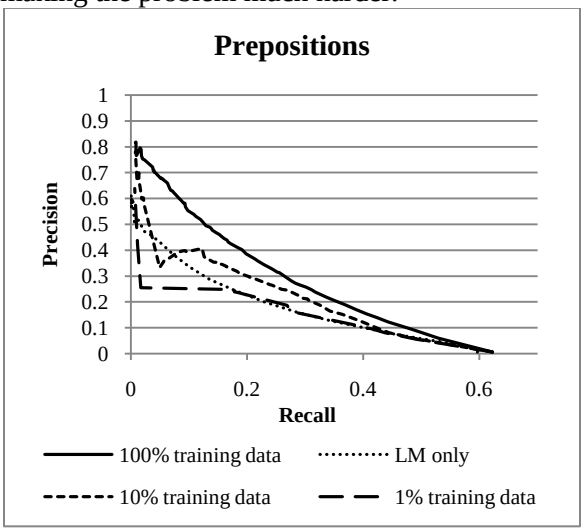

Figure 3: Using different amounts of annotated training data for the preposition meta-classifier.

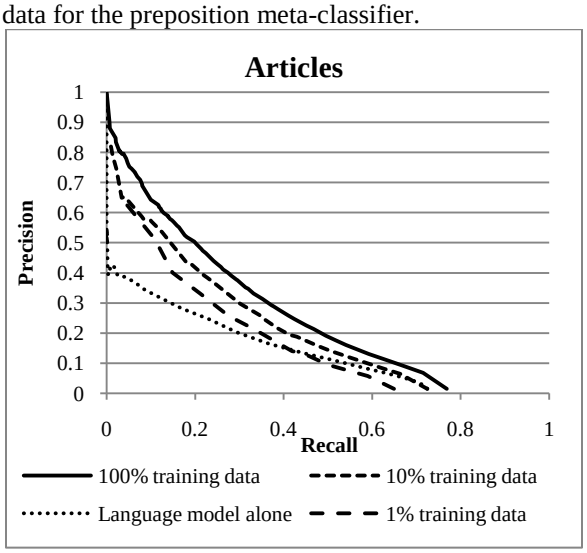

Figure 4: Using different amounts of annotated training data for the article meta-classifier.

To find out if it is possible to reduce the required amount of annotated preposition errors for a system that still covers more than one third of the preposition errors, we ran the same learning curve experiments but now only taking the four most frequent prepositions into account: *to, of, in, for*. In the CLC, these four prepositions account for 39.8% of preposition error flags. As in the previous experiments, however, we found that we are not able to outperform the baseline by using just 1% of annotated data.

5 Error Analysis

We have conducted a failure analysis on examples where the system produces a blatantly bad suggestion in order to see whether this decision could be attributed to the error-specific classifier or to the language model, or both, and what the underlying cause is. This preliminary analysis highlights two common causes for bad flags. One is that of frequent lower order n-grams that dominate the language model score. Consider the CLEC sentence *I get to know the world outside the campus **by** newspaper and television*. The system suggests deleting *by*. The cause of this bad decision is that the bigram *campus newspaper* is extremely likely, trumping all other n-grams, and leading to a high probability for the suggested string compared to the original: Log (P(original)) = -26.2 and Log (P(suggestion)) = -22.4. This strong imbalance of the language model score causes the meta-classifier to assign a relatively high probability to this being a correct revision, even though the error-specific classifier is on the right track and gives a relatively high probability for the presence of a preposition and the choice of *by*. A similar example, but for substitution, occurs in *They give discounts to their workers **on** books.* Here the bigram *in books* has a very high probability and the system incorrectly suggests replacing *on* with *in*. An example for insertion is seen in *Please send me the letter **back** writing what happened.* Here, the bigram *back to* causes the bad suggestion of inserting *to* after *back*. Since the language model is generally more accurate than the error-specific classifier, the meta-classifier tends to trust its score more than that of the classifier. As a result we see this kind of error quite frequently.

Another common error class is the opposite situation: the language model is on the right track, but the classifier makes a wrong assessment. Consider *Whatever direction my leg fought to stretch*, with the suggested insertion of *on* before *my leg*. Here Log (P(original)) = -31.5 and Log (P(suggestion)) = -32.1, a slight preference for the original string. The error-specific classifier, however, assigns a probability of 0.65 for a preposition to be present, and 0.80 for that preposition to be *on*. The contextual features that are important in that decision are: the insertion site is between a pronoun and a noun, it is relatively close to the beginning of the sentence, and the head of the NP *my leg* has a possible

mass noun sense. An example involving deletion is in *Someone came to sort **of** it.* While the language model assigns a high probability for deleting *of*, the error-specific classifier does not. Similarly, for substitution, in *Your experience is very interesting **for** our company*, the language model suggests substituting *for* with *to* while the classifier gives the substitution a very low probability.

As can be seen from the learner sentences cited above, often, even though the sentences are grammatical, they are not idiomatic, which can confuse all of the classifiers.

6 Conclusion and Future Work

We have addressed two questions in this paper:

1. How effective is a meta-classification approach that combines language modeling and error-specific classification to the detection and correction of preposition and article errors by non-native speakers?
2. How much error-annotated data is sufficient to produce positive results using that approach?

We have shown that a meta-classifier approach outperforms using a language model or a classifier alone. An interesting side result is that the language model solidly outperforms the contextual classifier for both article and preposition correction, contrary to current practice in the field. Training data requirements for the meta-classifier vary significantly between article and preposition error detection. The article meta-classifier can be trained with as few as 600 annotated errors, but the preposition meta-classifier requires more annotated data by an order of magnitude. Still, the overall amount of expensive error-annotated data is relatively small, and the meta-classification approach makes it possible to leverage large amounts of well-formed text in the primary models, tuning to the non-native domain in the meta-classifier.

We believe that the logical next step is to combine more primary models in the meta-classifier. Candidates for additional primary models include (1) more classifiers trained either on different data sets or with a different classification algorithm, and (2) more language models, such as skip models or part-of-speech n-gram language models.

Acknowledgments

We thank Claudia Leacock from the Butler Hill Group for detailed error analysis and the anonymous reviewers for helpful and constructive feedback.

References

Eric Steven Atwell. 1987. How to detect grammatical errors in a text without parsing it. In *Proceedings of the 3rd EACL* (pp 38 – 45). Copenhagen.

John Bitchener, Stuart Young, and Denise Cameron. 2005. The effect of different types of corrective feedback on ESL student writing. *Journal of Second Language Writing*, *14*(3), 191-205.

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

Martin Chodorow, Joel Tetreault, and Na-Rae Han. 2007. Detection of grammatical errors involving prepositions. In *Proceedings of the Fourth ACL-SIGSEM Workshop on Prepositions* (pp. 25-30). Prague.

Gerard M. Dalgish. 1985. Computer-assisted ESL research and courseware development. *Computers and Composition*, *2*(4), 45-62.

Rachele De Felice and Stephen G. Pulman. 2007. Automatically acquiring models of preposition use. In *Proceedings of the Fourth ACL-SIGSEM Workshop on Prepositions* (pp. 45-50). Prague.

Rachele De Felice and Stephen Pulman. 2008. A classifier-based approach to preposition and determiner error correction in L2 English. In *Proceedings of COLING*. Manchester, UK.

Thomas G. Dietterich. 1997. Machine learning research: Four current directions. *AI Magazine*, *18*(4), 97-136.

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

Michael Gamon, Claudia Leacock, Chris Brockett, William B. Dolan, Jianfeng Gao, Dmitriy Belenko, and Alexandre Klementiev,. 2009. Using statistical techniques and web search to correct ESL errors. *CALICO Journal*, *26*(3).

Michael Gamon, Jianfeng Gao, Chris Brockett, Alexander Klementiev, William Dolan, Dmitriy Belenko, and Lucy Vanderwende. 2008. Using contextual speller techniques and language modeling for ESL error correction. In *Proceedings of IJCNLP*, Hyderabad, India.

Jianfeng Gao, Joshua Goodman, and Jiangbo Miao. 2001. The use of clustering techniques for language modeling—Application to Asian languages. *Compu-*

tational Linguistics and Chinese Language Processing, *6*(1), 27-60.

Andrew Golding. 1995. A Bayesian Hybrid for Context Sensitive Spelling Correction. In *Proceedings of the 3rd Workshop on Very Large Corpor*a (pp. 39–53). Cambridge, USA.

Andrew R. Golding and Dan Roth. 1996. Applying Winnow to context-sensitive spelling correction. In *Proceedings of the Int. Conference on Machine Learning* (pp 182 –190).

Na-Rae Han, Martin Chodorow, and Claudia Leacock. 2004. Detecting errors in English article usage with a maximum entropy classifier trained on a large, diverse corpus. In *Proceedings of the 4th International Conference on Language Resources and Evaluation.* Lisbon.

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

Trude Heift and Mathias Schulze. 2007. *Errors and Intelligence in Computer-Assisted Language Learning: Parsers and Pedagogues*. New York & London: Routledge.

Matthieu Hermet, Alain Désilets, and Stan Szpakowicz. 2008. Using the web as a linguistic resource to automatically correct lexico-yyntactic errors. In *Proceedings of the 6th Conference on Language Resources and Evaluation (LREC)*, (pp. 874 - 878).

Emi Izumi, Kiyotaka Uchimoto, Toyomi Saiga, Thepchai Supnithi and Hitoshi Isahara. 2003. Automatic error detection in the Japanese learners' English spoken data. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* (pp. 145-148).

Emi Izumi, Kiyotaka Uchimoto and Hitoshi Isahara. 2004. SST speech corpus of Japanese learners' English and automatic detection of learners' errors. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC)*, (Vol 4, pp. 31-48).

Kevin Knight and Ishwar Chander,. 1994. Automatic postediting of documents. In *Proceedings of the 12th National Conference on Artificial Intelligence* (pp. 779-784). Seattle: Morgan Kaufmann.

Claudia Leacock, Michael Gamon, and Chris Brockett. 2009. User Input and Interactions on Microsoft ESL Assistant. In *Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications* (pp. 73-81).

John Lee. 2004. Automatic article restoration. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, (*pp. 31-36). Boston.

Guido Minnen, Francis Bond, and Anne Copestake. 2000. Memory-based learning for article generation. In *Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop* (pp. 43-48). Lisbon.

Patrick Nguyen, Jianfeng Gao, and Milind Mahajan. 2007. MSRLM: A scalable language modeling toolkit. *Microsoft Technical Report* 2007-144. Redmond.

Adwait Ratnaparkhi. 1997. A simple introduction to maximum entropy models for natural language processing. *Technical Report IRCS Report 97-98*, Institute for Research in Cognitive Science, University of Pennsylvania.

Herman Stehouwer and Menno van Zaanen. 2009. Language models for contextual error detection and correction. In *Proceedings of the EACL 2009 Workshop on Computational Linguistic Aspects of Grammatical Inference* (pp. 41-48). Athens.

Joel Tetreault and Martin Chodorow. 2008a. The ups and downs of preposition error detection in ESL. In *Proceedings of COLING*. Manchester, UK.

Joel Tetreault and Martin Chodorow. 2008b. Native judgments of non-native usage: Experiments in preposition error detection. In *Proceedings of the Workshop on Human Judgments in Computational Linguistics, 22nd International Conference on Computational Linguistics* (pp 43-48). Manchester, UK.

Jenine Turner and Eugene Charniak. 2007. Language modeling for determiner selection. In *Human Language Technologies 2007: NAACL*; Companion Volume, Short Papers (pp. 177-180). Rochester, NY.

Wikipedia. English Language. http://en.wikipedia.org/wiki/English_language

Xing Yi, Jianfeng Gao, and Bill Dolan. 2008. A web-based English proofing system for English as a second language users. In *Proceedings of the Third International Joint Conference on Natural Language Processing (IJCNLP)*. Hyderabad, India.