Generating Confusion Sets for Context-Sensitive Error Correction

Alla Rozovskaya and Dan Roth
University of Illinois at Urbana-Champaign
Urbana, IL 61801
{rozovska,danr}@illinois.edu

Abstract

In this paper, we consider the problem of generating candidate corrections for the task of correcting errors in text. We focus on the task of correcting errors in preposition usage made by non-native English speakers, using discriminative classifiers. The standard approach to the problem assumes that the set of candidate corrections for a preposition consists of all preposition choices participating in the task. We determine likely preposition confusions using an annotated corpus of non-native text and use this knowledge to produce smaller sets of candidates.

We propose several methods of restricting candidate sets. These methods exclude candidate prepositions that are not observed as valid corrections in the annotated corpus and take into account the likelihood of each preposition confusion in the non-native text. We find that restricting candidates to those that are observed in the non-native data improves both the precision and the recall compared to the approach that views all prepositions as possible candidates. Furthermore, the approach that takes into account the likelihood of each preposition confusion is shown to be the most effective.

1 Introduction

We address the problem of generating candidate corrections for the task of correcting context-dependent mistakes in text, mistakes that involve confusing valid words in a language. A well-studied instance of this problem – context-sensitive spelling errors – has received a lot of attention in natural language research (Golding and Roth, 1999; Carlson et al., 2001; Carlson and Fette, 2007; Banko and Brill, 2001). The context-sensitive spelling correction task addresses the problem of correcting spelling mistakes that result in legitimate words, such as confusing *their* and *there* or *your* and *you're*. In this task, a *candidate set* or a *confusion set* is defined that specifies a list of confusable words, e.g., {*their, there*} or {*cite, site, sight*}. Each occurrence of a confusable word in text is represented as a vector of features derived from a small context window around the target. A classifier is trained on text assumed to be error-free, replacing each target word occurrence (e.g. *their*) with a *confusion set* consisting of {*their, there*}, thus generating both positive and negative examples, respectively, from the same context. Given a text to correct, for each word in text that belongs to the confusion set the classifier predicts the most likely candidate in the confusion set.

More recently, work in error correction has taken an interesting turn and focused on correcting errors made by English as a Second Language (ESL) learners, with a special interest given to errors in article and preposition usage. These mistakes are some of the most common mistakes for non-native English speakers of all proficiency levels (Dalgish, 1985; Bitchener et al., 2005; Leacock et al., 2010). Approaches to correcting these mistakes have adopted the methods of the context-sensitive spelling correction task. A system is usually trained on well-formed native English text (Izumi et al., 2003; Eeg-Olofsson and Knuttson, 2003; Han et al., 2006; Felice and Pulman, 2008; Gamon et al., 2008; Tetreault

Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 961–970,
MIT, Massachusetts, USA, 9-11 October 2010.

and Chodorow, 2008; Elghaari et al., 2010; Tetreault et al., 2010), but several works incorporate into training error-tagged data (Gamon, 2010; Han et al., 2010) or error statistics (Rozovskaya and Roth, 2010b). The classifier is then applied to non-native text to predict the correct article/preposition in context. The possible candidate selections include the set of all articles or all prepositions.

While in the article correction task the candidate set is small (*a*, *the*, no article), systems for correcting preposition errors, even when they consider the most common prepositions, may include between 9 to 34 preposition classes. For each preposition in the non-native text, every other candidate in the confusion set is viewed as a potential correction. This approach, however, does not take into account that writers do not make mistakes randomly: Not all candidates are equally likely given the preposition chosen by the author and errors may depend on the first language (L1) of the writer. In this paper, we define *L1-dependent candidate sets* for the preposition correction task (Section 4.1). L1-dependent candidate sets reflect preposition confusions observed with the speakers of the first language L1. We propose methods of enforcing L1-dependent candidate sets in training and testing.

We consider mistakes involving the top ten English prepositions. As our baseline system, we train a multi-class classifier in *one-vs-all* approach, which is a standard approach to multi-class classification. In this approach, a separate binary classifier for each preposition p_i, $1 \leq i \leq 10$, is trained, s.t. all p_i examples are positive examples for the classifier and all other nine classes act as negative examples. Thus, for each preposition p_i in non-native text there are ten[1] possible prepositions that the classifier can propose as corrections for p_i.

We contrast this baseline method to two methods that enforce L1-dependent candidate sets in training. First, we train a separate classifier for each preposition p_i on the prepositions that belong to *L1-dependent candidate set* of p_i. In this setting, the negative examples for p_i are those that belong to *L1-dependent candidate set* of p_i.

The second method of enforcing *L1-dependent candidate sets* in training is to train on native data with artificial preposition errors in the spirit of Rozovskaya and Roth (2010b), where the errors mimic the error rates and error patterns of the non-native text. This method requires more knowledge, since it uses a distribution of errors from an error-tagged corpus.

We also propose a method of enforcing *L1-dependent candidate sets* in testing, through the use of a confidence threshold. We consider two ways of applying a threshold: (1) the standard way, when a correction is proposed only if the classifier's confidence is sufficiently high and (2) L1-dependent threshold, when a correction is proposed only if it belongs to *L1-dependent candidate set*.

We show that the methods of restricting candidate sets to L1-dependent confusions improve the preposition correction system. We demonstrate that restricting candidate sets to those prepositions that are confusable in the data by L1 writers is beneficial, when compared to a system that assumes an unrestricted candidate set by considering as valid corrections all prepositions participating in the task. Furthermore, we find that the most effective method is the one that uses knowledge about the likelihoods of preposition confusions in the non-native text introduced through artificial errors in training.

The rest of the paper is organized as follows. First, we describe related work on error correction. Section 3 presents the ESL data and statistics on preposition errors. Section 4 describes the methods of restricting candidate sets in training and testing. Section 5 describes the experimental setup. We present and discuss the results in Section 6. The key findings are summarized in Table 5 and Fig. 1 in Section 6. We conclude with a brief discussion of directions for future work.

2 Related Work

Work in text correction has focused primarily on correcting context-sensitive spelling errors (Golding and Roth, 1999; Banko and Brill, 2001; Carlson et al., 2001; Carlson and Fette, 2007) and mistakes made by ESL learners, especially errors in article and preposition usage.

Roth (1998) takes a unified approach to resolving semantic and syntactic ambiguities in natural lan-

[1]This includes the preposition p_i itself. If proposed by the classifier, it would not be flagged as an error.

guage by treating several related problems, including word sense disambiguation, word selection, and context-sensitive spelling correction as instances of the disambiguation task. Given a *candidate set* or a *confusion set* of confusable words, the task is to select the most likely candidate in context. Examples of confusion sets are {*sight, site, cite*} for context-sensitive spelling correction, {*among, between*} for word selection, or a set of prepositions for the preposition correction problem.

Each occurrence of a candidate word in text is represented as a vector of features. A classifier is trained on a large corpus of error-free text. Given text to correct, for each word in text that belongs to the confusion set the classifier is used to predict the most likely candidate in the confusion set given the word's context.

In the same spirit, models for correcting ESL errors are generally trained on well-formed native text. Han et al. (2006) train a maximum entropy model to correct article mistakes. Chodorow et. al (2007), Tetreault and Chodorow (2008), and De Felice and Pulman (2008) train a maximum entropy model and De Felice and Pulman (2007) train a voted perceptron algorithm to correct preposition errors. Gamon et al. (2008) train a decision tree model and a language model to correct errors in article and preposition usage. Bergsma et al. (2009) propose a Naïve Bayes algorithm with web-scale N-grams as features, for preposition selection and context-sensitive spelling correction.

The set of valid candidate corrections for a target word includes all words in the confusion set. For the preposition correction task, the entire set of prepositions considered for the task is viewed as the set of possible corrections for each preposition in non-native text. Given a preposition with its surrounding context, the model selects the most likely preposition from the set of all candidates, where the set of candidates consists of nine (Felice and Pulman, 2008), 12 (Gamon, 2010), or 34 (Tetreault et al., 2010; Tetreault and Chodorow, 2008) prepositions.

2.1 Using Error-tagged Data in Training

Several recent works explore ways of using annotated non-native text when training error correction models.

One way to incorporate knowledge about which confusions are likely with ESL learners into the error correction system is to train a model on error-tagged data. Preposition confusions observed in the non-native text can then be included in training, by using the preposition chosen by the author (the *source* preposition) as a feature. This is not possible with a system trained on native data, because each *source* preposition is always the correct preposition.

Han et al. (2010) train a model on partially annotated Korean learner data. The error-tagged model trained on one million prepositions obtains a slightly higher recall and a significant improvement in precision (from 0.484 to 0.817) over a model fives times larger trained on well-formed text.

Gamon (2010) proposes a hybrid system for preposition and article correction, by incorporating the scores of a language model and class probabilities of a maximum entropy model, both trained on native data, into a meta-classifier that is trained on a smaller amount of annotated ESL data. The meta-classifier outperforms by a large margin both of the native models, but it requires large amounts of expensive annotated data, especially in order to correct preposition errors, where the problem complexity is much larger.

Rozovskaya and Roth (2010b) show that by introducing into native training data artificial article errors it is possible to improve the performance of the article correction system, when compared to a classifier trained on native data. In contrast to Gamon (2010) and Han et al. (2010) that use annotated data for training, the system is trained on native data, but the native data are transformed to be more like L1 data through artificial article errors that mimic the error rates and error patterns of non-native writers. This method is cheaper, since obtaining error statistics requires much less annotated data than training. Moreover, the training data size is not restricted by the amount of the error-tagged data available. Finally, the *source* article of the writer can be used in training as a feature, in the exact same way as with the models trained on error-tagged data, providing knowledge about which confusions are likely. Unlike article errors, preposition errors lend themselves very well to a study of confusion sets because the set of prepositions participating in the task is a lot bigger than the set of article choices.

3 ESL Data

3.1 Preposition Errors in Learner Data

Preposition errors are one of the most common mistakes that non-native speakers make. In the Cambridge Learner Corpus[2] (CLC), which contains data by learners of different first language backgrounds and different proficiency levels, preposition errors account for about 13.5% of all errors and occur on average in 10% of all sentences (Leacock et al., 2010). Similar error rates have been reported for other annotated ESL corpora, e.g. (Izumi et al., 2003; Rozovskaya and Roth, 2010a; Tetreault et al., 2010). Learning correct preposition usage in English is challenging for learners of all first language backgrounds (Dalgish, 1985; Bitchener et al., 2005; Gamon, 2010; Leacock et al., 2010).

3.2 The Annotated Corpus

We use data from an annotated corpus of essays written by ESL students. The essays were fully corrected and error-tagged by native English speakers. For each preposition used incorrectly by the author, the annotator also indicated the correct preposition choice. Rozovskaya and Roth (2010a) provide a detailed description of the annotation of the data.

The annotated data include sentences by speakers of five first language backgrounds: Chinese, Czech, Italian, Russian, and Spanish. The Czech, Italian, Russian and Spanish data come from the International Corpus of Learner English (ICLE, (Granger et al., 2002)), which is a collection of essays written by advanced learners of English. The Chinese data is a part of the Chinese Learners of English corpus (CLEC, (Gui and Yang, 2003)) that contains essays by students of all levels of proficiency. Table 1 shows preposition statistics based on the annotated data.

The combined data include 4185 prepositions, 8.4% of which were judged to be incorrect by the annotators. Table 1 demonstrates that the error rates in the Chinese speaker data, for which different proficiency levels are available, are 2 or 3 times higher than the error rates in other language groups. The data for other languages come from very advanced learners and, while there are also proficiency differences among advanced speakers, their error rates are much lower.

Source language	Total preps.	Incorrect preps.	Error rate
Chinese	953	144	15.1%
Czech	627	28	4.5%
Italian	687	43	6.3%
Russian	1210	85	7.0%
Spanish	708	52	7.3%
All	4185	352	8.4%

Table 1: **Statistics on prepositions in the ESL data.** Column *Incorrect* denotes the number of prepositions judged to be incorrect by the native annotators. Column *Error rate* denotes the proportion of prepositions used incorrectly.

We would also like to point out that we take as the *baseline*[3] for the task the accuracy of the non-native data, or the proportion of prepositions used correctly. Using the error rate numbers shown in Table 1, the baseline for Chinese speakers is thus 84.9%, and for all the data combined it is 91.6%.

3.3 Preposition Errors and L1

We focus on *preposition confusion* errors, mistakes that involve an incorrectly selected preposition[4]. We consider ten most frequent prepositions in English: *on, from, for, of, about, to, at, in, with,* and *by*[5].

We mentioned in Section 2 that not all preposition confusions are equally likely to occur and preposition errors may depend on the first language of the writer. Han et al. (2010) show that preposition errors in the annotated corpus by Korean learners are not evenly distributed, some confusions occurring more often than others. We also observe that confusion frequencies differ by L1. This is consistent with other studies, which show that learners' errors are influenced by their first language (Lee and Seneff, 2008; Leacock et al., 2010).

[2] http://www.cambridge.org/elt

[3] It is argued in Rozovskaya and Roth (2010b) that the most frequent class baselines are not relevant for error correction tasks. Instead, the error rate in the data need to be considered, when determining the baseline.

[4] We do not address errors of missing or extraneous prepositions.

[5] It is common to restrict the systems that detect errors in preposition usage to the top prepositions. In the CLC corpus, the usage of the ten most frequent prepositions accounts for 82% of all preposition errors (Leacock et al., 2010).

4 Methods of Improving Candidate Sets

In this section, we describe methods of restricting candidate sets according to the first language of the writer. For the preposition correction task, the standard approach considers all prepositions participating in the task as valid corrections for every preposition in the non-native data.

In Section 3.3, we pointed out that (1) not all preposition confusions are equally likely to occur and (2) preposition errors may depend on the first language of the writer. The methods of restricting confusion sets proposed in this work use knowledge about which prepositions are confusable based on the data by speakers of language L1.

We refer to the preposition originally chosen by the author in the non-native text as the *source* preposition, and *label* denotes the correct preposition choice, as chosen by the annotator. Consider, for example, the following sentences from the annotated corpus.

1. We ate **by*/with** our hands .
2. To tell the truth , time spent in jail often changes prisoners **to*/for** the worse.
3. And the problem that immediately appeared was that men were unable to cope **with** the new woman image .

In example 1, the annotator replaced *by* with *with*; *by* is the *source* preposition and *with* is the *label*. In example 2, *to* is the *source* and *for* is the *label*. In example 3, the preposition *with* is judged as correct. Thus, *with* is both the *source* and the *label*.

4.1 L1-Dependent Confusion Sets

Let source preposition p_i denote a preposition that appears in the data by speakers of L1. Let *ConfSet* denote the set of all prepositions that the system can propose as a correction for source preposition p_i. We define two types of confusion sets *ConfSet*. An unrestricted confusion set *AllConfSet* includes all ten prepositions. *L1-dependent confusion set L1ConfSet*(p_i) is defined as follows:

Definition $L1ConfSet(p_i) = \{p_j | \exists$ a sentence in which an L1 writer replaced preposition p_j with $p_i \}$

For example, in the Spanish speaker data, *from* is used incorrectly in place of *of* and *for*. Then for Spanish speakers, *L1ConfSet*(from)={from, of, for}.

Source prep. p_i	***L1ConfSet***(p_i)
on	{on, about, of, to, at, in, with, by}
by	{with, by, in}
from	{of, from, for}

Table 2: L1-dependent confusion sets for three prepositions based on data by Chinese speakers.

Table 2 shows for Chinese speakers three prepositions and their L1-dependent confusion sets.

We now describe methods of enforcing *L1-dependent confusion sets* in training and testing.

4.2 Enforcing L1-dependent Confusion Sets in Training

We propose two methods of enforcing L1-dependent confusion sets in training. They are contrasted to the typical method of training a multi-class 10-way classifier, where each class corresponds to one of the ten participating prepositions.

First, we describe the typical training setting.

NegAll Training proceeds in a standard way of training a multi-class classifier (*one-vs-all* approach) on all ten prepositions using well-formed native English data. For each preposition p_i, p_i examples are positive and the other nine prepositions are negative examples.

We now describe two methods of enforcing L1-dependent confusion sets in training.

NegL1 This method explores the difference between training with nine types as negative examples and (fewer than nine) L1-dependent negative examples.

For every preposition p_i, we train a classifier using only examples that are in *L1ConfSet*(p_i). In contrast to *NegAll*, for each source preposition, the negative examples are not all other nine types, but only those that belong in *L1ConfSet*(p_i). For each language L1, we train ten classifiers, one for each source preposition. For source preposition p_i in test, we consult the classifier for p_i. In this model, the confusion set for source p_i is restricted through training, since for source p_i, the possible candidate replacements are only those that the classifier sees in training, and they are all in *L1ConfSet*(p_i).

Training data	Negative examples	
	NegAll	**NegL1**
Clean	NegAll-Clean	NegL1-Clean
ErrorL1	NegAll-ErrorL1	-

Table 3: Training conditions that result in unrestricted (All) and L1-dependent training paradigms.

ErrorL1 This method restricts the candidate set to *L1ConfSet*(p_i) by generating artificial preposition errors in the spirit of Rozovskaya and Roth (2010b). The training data are thus no longer well-formed or *clean*, but augmented with L1 error statistics. Specifically, each preposition p_i in training is replaced with a different preposition p_j with probability *probConf*, s.t.

$$probConf = prob(p_i|p_j) \quad (1)$$

Suppose 10% of all source prepositions *to* in the Russian speaker data correspond to label *for*. Then *for* is replaced with *to* with probability 0.1.

The classifier uses in training the source preposition as a feature, which cannot be done when training on well-formed text, as discussed in Section 2.1. By providing the source preposition as a feature, we enforce L1-dependent confusion sets in training, because the system learns which candidate corrections occur with source preposition p_i. An important distinction of this approach is that it does not simply provide L1-dependent confusion sets in training: Because errors are generated using L1 writers' error statistics, the likelihood of each candidate correction is also provided. This approach is also more knowledge-intensive, as it requires annotated data to obtain error statistics.

It should be noted that this method is orthogonal to the *NegAll* and *NegL1* methods of training described above and can be used in conjunction with each of them, only that it transforms the training data to account in a more natural way for ESL writing.

We combine the proposed methods *NegAll*, *NegL1* with the *Clean* or *ErrorL1* methods and create three training approaches shown in Table 3.

4.3 Restricting Confusion Sets in Testing

To reduce the number of false alarms, correction systems generally use a threshold on the confidence of the classifier, following (Carlson et al., 2001), and propose a correction only when the confidence of the classifier is above the threshold. We show in Section 5 that the system trained on data with artificial errors performs competitively even without a threshold. The other systems use a threshold. We consider two ways of applying a threshold[6]:

1. ThreshAll A correction for source preposition p_i is proposed only when the confidence of the classifier exceeds the threshold. For each preposition in the non-native data, this method considers *all* candidates as valid corrections.

2. ThreshL1Conf A correction for source preposition p_i is proposed only when the confidence of the classifier exceeds the empirically found threshold and the preposition proposed as a correction for p_i is in the confusion set *L1ConfSet*(p_i).

5 Experimental Setup

In this section, we describe experiments with L1-dependent confusion sets. Combining the three training conditions shown in Table 3 with the two ways of thresholding described in Section 4.3, we build four systems[7]:

1. NegAll-Clean-ThreshAll This system assumes both in training and in testing stages that all preposition confusions are possible. The system is trained as a multi-class 10-way classifier, where for each preposition p_i, all other nine prepositions are negative examples. In testing, when applying the threshold, all prepositions are considered as valid corrections.

2. NegAll-Clean-ThreshL1 This system is trained exactly as *NegAll-Clean-ThreshAll* but in testing only corrections that belong

[6]Thresholds are found empirically: We divide the evaluation data into three equal parts and to each part apply the threshold, which is optimized on the other two parts of the data.

[7]In testing, it is not possible to consider a confusion set larger than the one used in training. Therefore, *ThreshAll* is only possible with *NegAll* training condition.

to *L1ConfSet*(p_i) are considered as valid corrections for p_i.

3. **NegL1-Clean-ThreshL1** For each preposition p_i, a separate classifier is trained on the prepositions that are in *L1ConfSet*(p_i), where p_i examples are positive and a set of (fewer than nine) p_i-dependent prepositions are negative. Only corrections that belong to *L1ConfSet*(p_i) are considered as valid corrections for p_i.[8] Ten p_i-dependent classifiers for each L1 are trained.

4. **NegAll-ErrorL1-NoThresh** A system is trained as a multi-class 10-way classifier with artificial preposition errors that mimic the errors rates and confusion patterns of the non-native text. For each L1, an L1-dependent system is trained. This system does not use a threshold. We discuss this in more detail below.

The system *NegAll-Clean-ThreshAll* is our *baseline* system. It assumes both in training and in testing that all preposition confusions are possible.

All of the systems are trained on the same set of word and part-of-speech features using the same set of training examples. Features are extracted from a window of eight words around the preposition and include words, part-of-speech tags and conjunctions of words and tags of lengths two, three, and four. Training data are extracted from English Wikipedia and the New York Times section of the Gigaword corpus (Linguistic Data Consortium, 2003).

In each training paradigm, we follow a discriminative approach, using an online learning paradigm and making use of the Averaged Perceptron Algorithm (Freund and Schapire, 1999) – we use the regularized version in Learning Based Java[9] (LBJ, (Rizzolo and Roth, 2007)). While classical Perceptron comes with generalization bound related to the margin of the data, Averaged Perceptron also comes with a PAC-like generalization bound (Freund and Schapire, 1999). This linear learning algorithm is known, both theoretically and experimentally, to be among the best linear learning approaches and is competitive with SVM and Logistic Regression, while being more efficient in training. It also has been shown to produce state-of-the-art results on many natural language applications (Punyakanok et al., 2008).

[8] *ThreshAll* is not possible with this training option, as the system never proposes a correction that is not in *L1ConfSet*(p_i).

[9] LBJ code is available at `http://cogcomp.cs.illinois.edu/page/software`

6 Results and Discussion

Table 4 shows performance of the four systems by the source language. For each source language, the methods that restrict candidate sets in training or testing outperform the baseline system *NegAll-Clean-ThreshAll* that does not restrict candidate sets. The *NegAll-ErrorL1-NoThresh* system performs better than the other three systems for all languages, except for Italian. In fact, for the Czech speaker data, all systems other than *NegAll-ErrorL1-NoThresh*, have a precision and a recall of 0, since no errors are detected[10].

Source lang.	System	Acc.	P	R
CH	*NegAll-Clean-ThreshAll*	84.78	47.58	11.46
	NegAll-Clean-ThreshL1	84.84	48.05	15.28
	NegL1-Clean-ThreshL1	84.94	50.87	11.46
	NegAll-ErrorL1-NoThresh	86.36	**55.27**	**27.43**
	`Baseline`	84.89		
CZ	*NegAll-Clean-ThreshAll*	94.74	0.00	0.00
	NegAll-Clean-ThreshL1	94.98	0.00	0.00
	NegL1-Clean-ThreshL1	94.66	0.00	0.00
	NegAll-ErrorL1-NoThresh	95.85	**75.00**	**10.71**
	`Baseline`	95.53		
IT	*NegAll-Clean-ThreshAll*	93.23	26.14	8.14
	NegAll-Clean-ThreshL1	94.03	**51.59**	**18.60**
	NegL1-Clean-ThreshL1	93.16	35.00	16.28
	NegAll-ErrorL1-NoThresh	93.60	44.95	10.47
	`Baseline`	93.74		
RU	*NegAll-Clean-ThreshAll*	92.73	31.11	3.53
	NegAll-Clean-ThreshL1	93.02	48.81	8.24
	NegL1-Clean-ThreshL1	92.44	34.42	8.82
	NegAll-ErrorL1-NoThresh	93.14	**52.38**	**12.94**
	`Baseline`	92.98		
SP	*NegAll-Clean-ThreshAll*	91.95	26.14	5.77
	NegAll-Clean-ThreshL1	92.02	28.64	5.77
	NegL1-Clean-ThreshL1	92.44	40.00	7.69
	NegAll-ErrorL1-NoThresh	93.71	**77.50**	**19.23**
	`Baseline`	92.66		

Table 4: **Performance results for the 4 systems**. All systems, except for *NegAll-ErrorL1-NoThresh*, use a threshold, which is optimized for accuracy on the development set. *Baseline* denotes the percentage of prepositions used correctly in the data. The baseline allows us to evaluate the systems with respect to accuracy, the percentage of prepositions, on which the prediction of the system is the same as the label. Averaged results over 2 runs.

[10] The Czech data set is the smallest and contains a total of 627 prepositions and only 28 errors.

The *NegAll-ErrorL1-NoThresh* system does not use a threshold. However, as shown in Fig. 1, it is possible to increase the precision of the *NegAll-ErrorL1-NoThresh* system by applying a threshold, at the expense of a lower recall.

While the ordering of the systems with respect to quality is not consistent from Table 4, due to modest test data sizes, Table 5 and Fig. 1 show results for the models on all data combined and thus give a better idea of how the systems compare against each other.

Table 5 shows performance results for all data combined. Both *NegAll-Clean-ThreshL1* and *NegL1-Clean-ThreshL1* achieve a better precision and recall over the system with an unrestricted candidate set *NegAll-Clean-ThreshAll*. Recall that both of the systems restrict candidate sets, the former at testing stage, the latter by training a separate classifier for each source preposition. *NegAll-Clean-ThreshL1* performs slightly better than *NegL1-Clean-ThreshL1*. We hypothesize that the *NegAll-Clean-ThreshAll* performance may be affected because the classifiers for different source prepositions contain different number of classes, depending on the size of *L1ConfSet* confusion sets, which makes it more difficult to find a unified threshold. The best performing system overall is *NegAll-ErrorL1-NoThresh*. While *NegAll-Clean-ThreshL1* and *NegL1-Clean-ThreshL1* restrict candidate sets, *NegAll-ErrorL1-NoThresh* also provides information about the likelihood of each confusion, which benefits the classifier. The differences between *NegAll-ErrorL1-ThreshL1* and each of the other three systems are statistically significant[11] (McNemar's test, $p < 0.01$). The table also demonstrates that the results on the correction task may vary widely. For example, the recall varies by language between 10.47% and 27.43% for the *NegAll-ErrorL1-NoThresh system*. The highest recall numbers are obtained for Chinese speakers. These speakers also have the highest error rate, as we noted in Section 3.

[11]Tests of statistical significance compare the combined results from all language groups for each model. For example, to compare the model *NegAll-Clean-ThreshAll* to *NegAll-ErrorL1-NoThresh*, we combine the results from the five language-specific models *NegAll-ErrorL1-NoThresh* and compare them to the results on the combined data from the five language groups achieved by the model *NegAll-Clean-ThreshAll*.

System	Acc.	P	R
NegAll-Clean-ThreshAll	90.90	31.11	7.95
NegAll-Clean-ThreshL1	91.11	37.82	12.78
NegL1-Clean-ThreshL1	90.97	34.34	9.66
NegAll-ErrorL1-NoThresh	92.23	**58.47**	**19.60**

Table 5: **Comparison of the performance of the 4 systems on all data combined**. All systems, except for *NegAll-ErrorL1-NoThresh*, use a threshold, which is optimized for accuracy on the development set. The differences between *NegAll-ErrorL1-ThreshL1* and each of the other three systems are statistically significant (McNemar's test, $p < 0.01$).

Finally, Fig. 1 shows precision/recall curves for the systems[12]. The curves are obtained by varying a decision threshold for each system. Before we examine the differences between the models, it should be noted that in error correction tasks precision is favored over recall due to the low level of error.

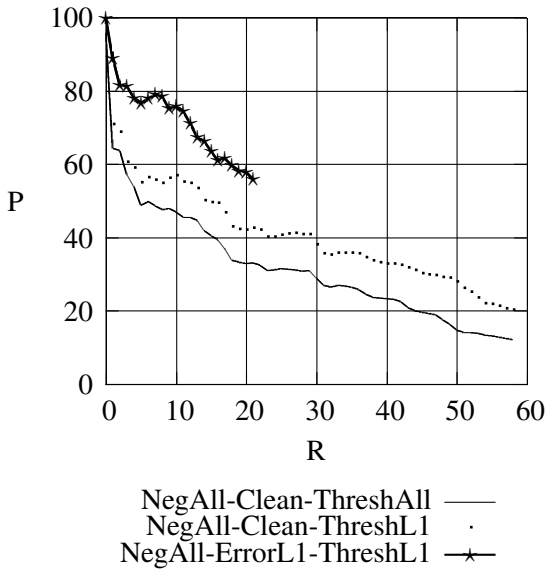

Figure 1: Precision and recall (%) for three models: *NegAll-Clean-ThreshAll*, *NegAll-Clean-ThreshL1*, and *NegAll-ErrorL1-ThreshL1*.

The curves demonstrate that *NegAll-Clean-ThreshL1* and *NegAll-ErrorL1-ThreshL1* are superior to the baseline system *NegAll-Clean-ThreshAll*: on the same recall points, the precision for both systems is consistently better than for the base-

[12]*NegL1-Clean-ThreshL1* is not shown, since it is similar in its behavior to *NegAll-Clean-ThreshL1*.

line model[13]. Moreover, while restricting candidate sets improves the results, providing information to the classifier about the likelihoods of different confusions is more helpful, which is reflected in the precision differences between *NegAll-Clean-ThreshL1* and *NegAll-ErrorL1-ThreshL1*. In fact, *NegAll-ErrorL1-ThreshL1* achieves a higher precision compared to the other systems, even when no threshold is used (Tables 4 and 5). This is because, unlike the other models, this system does not tend to propose too many false alarms.

6.1 Comparison to Other Systems

It is difficult to compare performance to other systems, since training and evaluation are not performed on the same data, and results may vary widely depending on the first language and proficiency level of the writer. However, in Table 6 we list several systems and their performance on the task. Tetreault et al. (2010) train on native data and obtain a precision of 48.6% and a recall of 22.5% with top 34 prepositions on essays from the Test of English as a Foreign Language exams. Han et al. (2010) obtain a precision of 81.7% and a recall of 13.2% using a model trained on partially error-tagged data by Korean speakers on top ten prepositions. A model trained on 2 million examples from clean text achieved on the same data set a precision of 46.3% and a recall of 11.6%.

Gamon (2010) shows precision/recall curves on the combined task of detecting missing, extraneous and confused prepositions. For recall points 10% and 20%, precisions of 55% and 40%, respectively, are obtained. For our data, a recall of 10% corresponds to a precision of 46% for the worst-performing model and 78% for the best-performing model. For 20% recall, we obtain a precision of 33% for the worst-performing model and 58% for the best-performing model. We would like to emphasize that these comparisons should be interpreted with caution.

[13]While significance tests did not show differences between *NegAll-Clean-ThreshAll* and *NegAll-Clean-ThreshL1*, perhaps due to a modest test set size, the curves demonstrate that the latter system indeed provides a stable advantage over the baseline unrestricted approach.

7 Conclusion and Future Work

In this paper, we proposed methods for improving candidate sets for the task of detecting and correcting errors in text. To correct errors in preposition usage made by non-native speakers of English, we proposed L1-dependent confusion sets that determine valid candidate corrections using knowledge about preposition confusions observed in the non-native text. We found that restricting candidates to those that are observed in the non-native data improves both the precision and the recall compared to a classifier that considers as possible candidates the set of all prepositions. Furthermore, the approach that takes into account the likelihood of each preposition confusion is shown to be the most effective.

System	Training Data	P	R
Tetreault et al., 2010	native; 34 preps.	48.6	22.5
Han et al., 2010	partially error-tagged; 10 preps.	81.7	13.2
Han et al., 2010	native; 10 preps.	46.3	11.6
Gamon, 2010	native; 12 preps.+ extraneous+missing	33.0	10.0
Gamon, 2010	native+error-tagged; 12 preps.+ extraneous+missing	55.0	10.0
NegAll-Clean-ThreshAll	native; 10 preps.	46.0	10.0
NegAll-ErrorL1-ThreshL1	native with L1 error statistics; 10 preps.	78.0	10.0

Table 6: **Comparison to other systems**. Please note that a direct comparison is not possible, since the systems are trained and evaluated on different data sets. Gamon (2010) also considers missing and extraneous preposition errors.

The methods proposed in this paper make use of select characteristics that the error-tagged data can provide. We would also like to compare the proposed methods to the quality of a model trained on error-tagged data. Improving the system is also in our future work, but orthogonal to the current contribution.

Acknowledgments

We thank Nick Rizzolo for helpful discussions on LBJ. We also thank Peter Chew and the anonymous reviewers for their insightful comments. This research is partly supported by a grant from the U.S. Department of Education.

References

M. Banko and E. Brill. 2001. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 26–33, Toulouse, France, July.

J. Bitchener, S. Young, and D. Cameron. 2005. The effect of different types of corrective feedback on ESL student writing. *Journal of Second Language Writing*.

A. Carlson and I. Fette. 2007. Memory-based context-sensitive spelling correction at web scale. In *Proceedings of the IEEE International Conference on Machine Learning and Applications (ICMLA)*.

A. J. Carlson, J. Rosen, and D. Roth. 2001. Scaling up context sensitive text correction. In *Proceedings of the National Conference on Innovative Applications of Artificial Intelligence (IAAI)*, pages 45–50.

M. Chodorow, J. Tetreault, and N.-R. Han. 2007. Detection of grammatical errors involving prepositions. In *Proceedings of the Fourth ACL-SIGSEM Workshop on Prepositions*, pages 25–30, Prague, Czech Republic, June. Association for Computational Linguistics.

G. Dalgish. 1985. Computer-assisted ESL research. *CALICO Journal*, 2(2).

J. Eeg-Olofsson and O. Knuttson. 2003. Automatic grammar checking for second language learners - the use of prepositions. *Nodalida*.

A. Elghaari, D. Meurers, and H. Wunsch. 2010. Exploring the data-driven prediction of prepositions in english. In *Proceedings of COLING 2010*, Beijing, China.

R. De Felice and S. Pulman. 2007. Automatically acquiring models of preposition use. In *Proceedings of the Fourth ACL-SIGSEM Workshop on Prepositions*, pages 45–50, Prague, Czech Republic, June.

R. De Felice and S. Pulman. 2008. A classifier-based approach to preposition and determiner error correction in L2 English. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 169–176, Manchester, UK, August.

Y. Freund and R. E. Schapire. 1999. Large margin classification using the perceptron algorithm. *Machine Learning*, 37(3):277–296.

M. Gamon, J. Gao, C. Brockett, A. Klementiev, W. Dolan, D. Belenko, and L. Vanderwende. 2008. Using contextual speller techniques and language modeling for ESL error correction. In *Proceedings of IJCNLP*.

M. Gamon. 2010. Using mostly native data to correct errors in learners' writing. In *NAACL*, pages 163–171, Los Angeles, California, June.

A. R. Golding and D. Roth. 1999. A Winnow based approach to context-sensitive spelling correction. *Machine Learning*, 34(1-3):107–130.

S. Granger, E. Dagneaux, and F. Meunier. 2002. *International Corpus of Learner English*. Presses universitaires de Louvain.

S. Gui and H. Yang. 2003. *Zhongguo Xuexizhe Yingyu Yuliaohu. (Chinese Learner English Corpus)*. Shanghai Waiyu Jiaoyu Chubanshe. (In Chinese).

N. Han, M. Chodorow, and C. Leacock. 2006. Detecting errors in English article usage by non-native speakers. *Journal of Natural Language Engineering*, 12(2):115–129.

N. Han, J. Tetreault, S. Lee, and J. Ha. 2010. Using an error-annotated learner corpus to develop and ESL/EFL error correction system. In *LREC*, Malta, May.

E. Izumi, K. Uchimoto, T. Saiga, T. Supnithi, and H. Isahara. 2003. Automatic error detection in the Japanese learners' English spoken data. In *The Companion Volume to the Proceedings of 41st Annual Meeting of the Association for Computational Linguistics*, pages 145–148, Sapporo, Japan, July.

C. Leacock, M. Chodorow, M. Gamon, and J. Tetreault. 2010. Morgan and Claypool Publishers.

J. Lee and S. Seneff. 2008. An analysis of grammatical errors in non-native speech in English. In *Proceedings of the 2008 Spoken Language Technology Workshop*.

V. Punyakanok, D. Roth, and W. Yih. 2008. The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2).

N. Rizzolo and D. Roth. 2007. Modeling Discriminative Global Inference. In *Proceedings of the First International Conference on Semantic Computing (ICSC)*, pages 597–604, Irvine, California, September. IEEE.

D. Roth. 1998. Learning to resolve natural language ambiguities: A unified approach. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 806–813.

A. Rozovskaya and D. Roth. 2010a. Annotating ESL errors: Challenges and rewards. In *Proceedings of the NAACL Workshop on Innovative Use of NLP for Building Educational Applications*.

A. Rozovskaya and D. Roth. 2010b. Training paradigms for correcting errors in grammar and usage. In *Proceedings of the NAACL-HLT*.

J. Tetreault and M. Chodorow. 2008. The ups and downs of preposition error detection in ESL writing. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 865–872, Manchester, UK, August.

J. Tetreault, J. Foster, and M. Chodorow. 2010. Using parse features for preposition selection and error detection. In *ACL*.