Correct Me If I'm Wrong: Fixing Grammatical Errors by Preposition Ranking

Roman Prokofyev, Ruslan Mavlyutov, Martin Grund, Gianluca Demartini,
and Philippe Cudré-Mauroux
eXascale Infolab,
University of Fribourg,
Switzerland
{firstname.lastname@unifr.ch}

ABSTRACT

The detection and correction of grammatical errors still represent very hard problems for modern error-correction systems. As an example, the top-performing systems at the preposition correction challenge CoNLL-2013 only achieved a F1 score of 17%. In this paper, we propose and extensively evaluate a series of approaches for correcting prepositions, analyzing a large body of high-quality textual content to capture language usage. Leveraging n-gram statistics, association measures, and machine learning techniques, our system is able to learn which words or phrases govern the usage of a specific preposition. Our approach makes heavy use of n-gram statistics generated from very large textual corpora. In particular, one of our key features is the use of n-gram association measures (e.g., Pointwise Mutual Information) between words and prepositions to generate better aggregated preposition rankings for the individual n-grams. We evaluate the effectiveness of our approach using cross-validation with different feature combinations and on two test collections created from a set of English language exams and StackExchange forums. We also compare against state-of-the-art supervised methods. Experimental results from the CoNLL-2013 test collection show that our approach to preposition correction achieves ∼30% in F1 score which results in 13% absolute improvement over the best performing approach at that challenge.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text analysis*; I.7.m [**Document and Text Processing**]: [Miscellaneous]

General Terms

Algorithms, Experimentation

CIKM'14, November 3–7, 2014, Shanghai, China.

ACM 978-1-4503-2598-1/14/11 ...$15.00.
http://dx.doi.org/10.1145/2661829.2661942.

Keywords

preposition correction; n-gram statistics; supervised learning; pointwise mutual information

1. INTRODUCTION

Grammatically correct textual content is highly valuable: On one hand, grammatically correct textual content is easier to understand by humans. On the other hand, automated systems (such as machine translation or speech recognition systems) can produce better results by integrating grammar correction techniques.

Compared to editorially curated content — such as news articles or e-commerce product pages — online User-Generated Content (UGC) is much more likely to contain grammatical errors. This is due to many factors, including the short time dedicated to content writing and proof-reading, but also to the fact that online authors may not be native speakers of the language they use to produce new content.

In this paper, we address the task of automatically correcting grammatical errors in textual content, focusing in particular on English language and on the problem of preposition correction. Prepositional errors have received relatively little attention by the research community, despite their importance. According to Leacock *et al.* [13], prepositional errors are the second most common error made by English learners and account for about 13% of all errors. Possible applications of our approach include correcting grammatical errors in a type-as-you-go fashion for Web applications, or curating textual content for further automated processing.

In this paper, we propose a set of approaches based on an n-gram decomposition of the input sentences. Specifically, our techniques indicate which preposition should most likely be used in a given sentence based on statistical evidence of words' associativity extracted from a large collection of English books. Our approach generates a ranked list of prepositions, which are ordered by their likelihood of being correct for the given query sentence.

In order to evaluate our proposed approach and to experimentally compare with state-of-the-art techniques, we rely both on standard evaluation collections as well as on a newly created dataset. Standard collections for this task are usually built from English language exams, which contain manually labeled errors made by non-native English

speakers. The new dataset that we create uses data from the Stack Exchange forum websites[1]. This second dataset is more challenging than commonly used collections since the language used on online social platforms is typically more informal than that of English exams or curated content.

In summary, the main contributions of this paper are:

- Novel features for preposition error correction based on n-gram statistics;
- Novel grammar correction approaches that operate at the sentence level;
- A new test collection to evaluate preposition correction over UGC;
- An experimental comparison of our approach against state-of-the-art supervised approaches using both standard and new test collections showing the effectiveness, robustness, and generality of our method.

The rest of this paper is structured as follows. We discuss related work in the area of preposition correction below in Section 2. Section 3 introduces the task we address, while Section 4 gives an overview of our system and describes how we leverage a large corpus of n-grams to detect and correct grammatical errors. In Section 5, we define the different features used by our approach and how they are combined by means of machine learning. We experimentally evaluate the different approaches we propose on several collections in Section 6, before concluding in Section 7.

2. RELATED WORK

Grammatical error correction is a popular task in the NLP community, where identifying and correcting wrong prepositions is studied as well. A recent initiative [5] introduces the task of preposition and determiner error detection and correction. The task is split into 3 parts—detection, recognition, and correction. The detection task is about determining if something is wrong in a text; The recognition task is about identifying the error and its type; The correction task, finally, is about proposing a correction that matches the gold standard. The approach we adopt in this work covers the three tasks as, given a sentence, it replaces some of the prepositions whenever it determines that the prepositions are wrong. We can thus evaluate our entire pipeline on the final output as compared to the gold standard corrections and compare our system against state-of-the-art approaches. In this initiative, the training collection consists of the Cambridge Learner Corpus (CLC) "First Certificate of English" (FCE) examinations of 2000 and 2001 created by Cambridge University. The two top-performing systems [4] and [12] respectively achieved 42%/17% and 61%/5% in terms of Precision/Recall. Both systems adopted multi-class classification approaches to choose the correct preposition from a predefined confusion set. In their work, the authors applied a sophisticated set of features, including lexical features, POS tags, head verbs and nouns, Web n-gram counts and word dependencies. The main difference between these two systems stems from using different preposition candidate sets. The system with lower recall and higher precision values used a set of only 11 most common prepositions, while the other system used a set of 36 prepositions. However, the test collection used in this challenge was not made publicly available, thus it is not possible for our research to compare with these results directly.

The more recent CoNLL-2013 Shared Task on grammatical error correction [16] organized by the National University of Singapore developed a new test collection from scratch using essays written by university students. The collection was made publicly available for further use and studies. In this work we compare our approach against the participating systems from this shared task. The top-performing systems in the preposition correction task used variations of statistical machine translation approaches and achieved a maximum F1 score of 17.5%, which is significantly less then the F1 score we achieve using the solution proposed in this paper. Other systems used either machine learning or language modeling approaches [11][19], including the ones based on statistical n-gram counts from large corpora, the most common one being the Google Web-1T corpus[2].

Another approach to grammatical error correction is to reduce the correction task to a language disambiguation task, following the assumption that the context of a preposition can completely determine the preposition itself. This allows to use high-quality texts as both training and test collections by simply omitting existing prepositions and choosing the potentially correct preposition based on its context solely. In this context, Bergsma *et al.* [1] applied unsupervised techniques for preposition selection by using the log-counts of the n-gram frequencies appearing in their context. In this case, the context consisted of a sliding window of n-grams around the preposition, where n was ranging from two to five. Additionally, the authors used supervised techniques with a very large training set[3] (100k+ samples) to learn linear combinations of the n-gram counts used in the unsupervised approach, observing minor effectiveness increase as compared to their unsupervised approach. Unfortunately, such approaches do not perform well on real-world error correction tasks since the number of incorrect prepositions is often much lower than the number of correct ones (e.g., around 5% of errors) in English learner collections. Thus, rather than simply omitting it, leveraging statistics about the existing preposition in a sentence becomes an important piece of evidence. In our work, we consider the original preposition and the probability of a writer confusing it with others.

Additional work in [6] shows that leveraging big-data n-gram statistics from the Web yields better performance compared to traditional linguistic features. In [8], Heilman *et al.* extend the approach discussed above [1] by complementing the n-gram log-count method with rule-based and supervised techniques. N-gram count statistics were also successfully applied to various Information Retrieval problems such as named entity recognition [18], query spelling correction and query segmentation [9]. Some of the most recent approaches to grammatical error correction integrate large collections of n-gram counts together with supervised [20] or unsupervised [10] techniques. As compared to such approaches, we also leverage a large n-gram corpus in our work but in addition focus on features like skip n-grams and n-gram distance.

[1] `http://stackexchange.com/`

[2] `https://catalog.ldc.upenn.edu/LDC2006T13`

[3] They used the New York Times (NYT) section of the Gigaword `http://catalog.ldc.upenn.edu/LDC2003T05` corpus.

We focus in this paper on correcting preposition usage at the sentence level. A recent approach proposed in [3] is able to analyze complete sentences and correct multiple errors that have interdependencies. Compared to this piece of work, the problem we address in this paper is more focused as it aims at correcting the usage of *prepositions*. Our focused approach allows us to obtain some significant effectiveness improvement (i.e., 13% absolute improvement) over state-of-the-art approaches for preposition correction.

As compared to previous work from the NLP field, our work tackles a more challenging issue, which is applying grammar correction to Web content rather than simply proposing error correction for academic texts produced by ESL writers.

3. TASK OVERVIEW

In this section, we define the problem we are tackling. Furthermore, we introduce the datasets we use in our evaluation and present the metrics we use to compare to previous systems.

3.1 Task Definition

In the following, we define the task we address in this work. Given a sentence in English that contains a preposition, our system generates a ranked list of prepositions that could be used in place of the original one. In case the top ranked preposition selected by our system matches the original preposition, our system performs no correction. In the other case, the system suggests a list of ranked prepositions as alternatives to the preposition used in original sentence by the author.

Formally, given a sentence $s = t_1..p_j..t_m$ consisting of a list of tokens t_i and a preposition p_j, and given a candidate preposition set $P = \{p_1, .., p_m\}$, the task of *preposition selection* consists in generating a list of prepositions from P ranked by their likelihood of being correct in order to potentially replace p_j. If the top ranked preposition is equal to p_j, then the sentence is considered correct by the approach. Otherwise, the the sentence is considered as incorrect and the top-1 preposition is selected instead. In this work, we use the set of the 49 most frequent English prepositions as our candidate preposition set P[4].

3.2 Training and Test Collections

Academic Test Collections.

As mentioned above in Section 2, there exist different training and test collections for grammatical error correction. Since we are addressing errors that people make when producing textual content, the most preferable option would be to reuse an existing collection of English learners' texts. The survey in [13] gives a comprehensive overview of available datasets of this type. Unfortunately, most of the exam correction datasets are proprietary and not publicly available.

However, as mentioned in Section 2, the CoNLL-2013 Shared Task on grammatical error correction published their test dataset consisting of 50 essays written by 25 non-native English students from the National University of Singapore. Thus, we are using *CoNLL-2013 dataset* as our primary test collection for results comparison. As a training collection, we decided to use the standard Cambridge First Certificate in English (FCE) examinations from 2000-2001, which was also permitted by the rules of the CoNLL-2013 Shared Task.

Stack Exchange Test Collection.

Most of the available datasets for this task were built using collections that were generated in an exam context, where learners know in advance that they need to do their best in terms of grammatical correctness. On the contrary, text written by non-native speakers on the Web usually contains many more grammatical errors because of the informal environment. Thus, in our work we want to additionally focus on correcting the actual errors made by people online. The Stack Exchange Q&A website network represents one example of UGC web sites, where users ask field-specific questions and others answer them.

With respect to our task, the Stack Exchange (SE) network possesses two very valuable properties:

- the network community is largely international, consisting of many non-native English speakers[5];
- the company running the SE network provides anonymized historical data dumps to the public[6].

The SE public dataset contains the edit history of every post (question or answer) from the SE network together with the comments accompanying the edits.

To create an evaluation collection of grammatical errors with corresponding corrections, we processed this dataset and extracted all edits that contained the string "grammar" in their comment field. For each extracted edit, we compared the original and the edited versions of the post and then only extracted those edits that modified one of the prepositions from our candidate preposition set.[7] The exact version of the SE data dump we used is from March 2013.

The statistics for the collections we are using in this work are summarized in Table 1. As compared to test collections based on academic text, where each document is manually checked by an expert, we do not have the full ground truth here as some of the errors might have been overlooked by the Web community. Therefore, we only extract sentences that have at least one error correction. Thus, the relative error percentage for the SE collection (38.2) is not directly comparable with the ones we observe in the academic collections.

As we mentioned earlier, we use the CoNLL-2013 dataset as a primary evaluation collection to compare the proposed approaches to the state-of-the-art, while we use the SE collection to determine if our proposed techniques are applicable on Web content as well.

3.3 Evaluation metrics

We evaluate our approaches in terms of Precision, Recall and F1 score, which are defined for grammatical error

[4]about, above, absent, across, after, against, along, alongside, amid, among, amongst, around, at, before, behind, below, beneath, beside, besides, between, beyond, but, by, despite, during, except, for, from, in, inside, into, of, off, on, onto, opposite, outside, over, since, than, through, to, toward, towards, under, underneath, until, upon, with.

[5]`https://www.quantcast.com/stackoverflow.com/geo/countries`

[6]`https://archive.org/details/stackexchange`

[7]The collection is available online at: `https://github.com/XI-lab/preposition-data-cikm2014`.

Table 1: Statistics of training and test collections. The first two are based on English language tests while the last one has been constructed based on StackExchange edit history.

Collection	# sentences	# PREPs	# PREP errors	% errors
Cambridge FCE collection	27119	60279	2883	4.8
CoNLL-2013 test collection	1375	3241	152	4.7
SE collection	5917	15814	6040	38.2

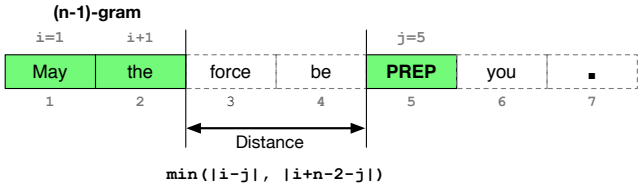

Figure 2: Tokenization example, PREP is a preposition placeholder.

correction as follows:

$$\text{Precision} = \frac{\text{valid suggested corrections}}{\text{total suggested corrections}}$$

$$\text{Recall} = \frac{\text{valid suggested corrections}}{\text{total valid corrections}}$$

$$\text{F1} = \frac{2 * Precision * Recall}{Precision + Recall}$$

4. SYSTEM OVERVIEW

4.1 Pipeline

The overall system for preposition correction works as follows (see Figure 1). Starting from a textual document, the system extracts sentences and tokenizes them. The generated list of tokens is used to produce n-grams, compute n-gram distances (Section 4.2), and to compute association measures based on external n-gram statistics (Section 4.3). These evidences are then used to produce features that allow a supervised classification component to select the most appropriate preposition in the context of a sentence (Section 4.5). Finally, the possibly corrected sentences are aggregated back into the document. Within this pipeline, in Section 5 we focus on how we use n-gram information to generate features for preposition ranking and selection.

4.2 N-Gram Distance

Given an input sentence, the preprocessing pipeline of our system consists of tokenizing the sentence and generating *n-grams* that contain pairs of *(n-1)-grams* and the preposition. To better understand the pipeline, we make use of the example tokenization in Figure 2 and Table 2. Here, the sentence is tokenized and we first consider all contiguous (n-1)-grams excluding the preposition itself. Second, we take these (n-1)-grams and generate all possible n-grams by adding the preposition, respecting the relative position of the preposition to the (n-1)-gram[8].

[8]If multiple prepositions occur in the same sentence, we form n-grams for each preposition independently.

Table 2: Example n-gram types and distances for the tokenization example on Figure 2 where $j = 5$.

N-gram	Type	Distance
the force PREP	3gram	-2
force be PREP	3gram	-1
be PREP you	3gram	0
PREP you .	3gram	1
be PREP	2gram	-1
PREP you	2gram	1
PREP .	2gram	2

Next, we define the *n-gram distance* of an n-gram containing a preposition based on the underlying (n-1)-gram used to generate it. In detail, given a tokenized sentence $\{t_1, .., t_m\}$ with a preposition in position j, we define the distance of an (n-1)-gram $[i, i + n - 2]$ as

$$dist([i, i+n-2]) = \begin{cases} C & \text{if } i > j \\ -C & \text{if } (i+n-2) < j \\ 0 & \text{else} \end{cases}$$

where $C = min(|i - j|, |i + n - 2 - j|)$.

Based on the definition of n-gram distance, we classify n-grams containing a preposition into three major classes based on their n-gram distance:

- *central n-grams* ($distance = 0$), where the preposition appears in the middle of an n-gram;
- *left-side n-grams* ($distance < 0$);
- *right-side n-grams* ($distance > 0$).

Note that in our work we consider punctuation marks as n-gram tokens; While for some applications punctuation needs to be filtered out, in our setting preposition usage can often be governed by their position with respect to a certain punctuation symbol.

4.3 N-gram association measures

As discussed in Section 2, most of the previous approaches tackled the preposition correction task by means of *preposition selection* via multi-class classification among preposition confusion sets. Contrary to this, we propose an approach to rank prepositions in a sentence according to some confidence score. This confidence score in turn incorporates information from the individual n-gram rankings of the prepositions.

The motivation behind our approach is that the usage of a certain preposition is often governed by a particular word

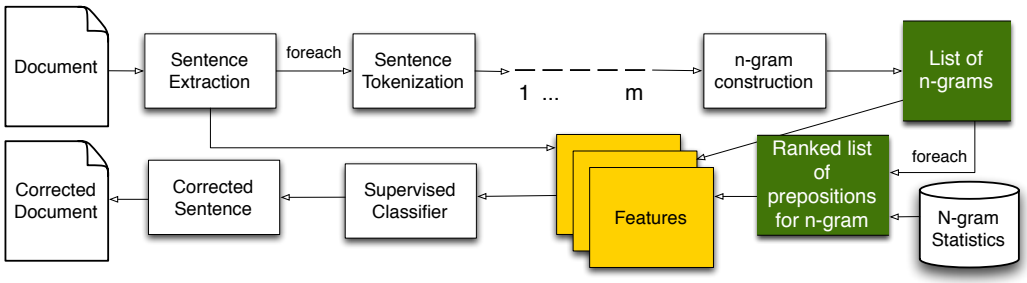

Figure 1: The pipeline of our preposition correction system. As input it receives documents to be corrected. Then, sentences are split and tokenized. Next, n-grams are generated, associative measures are computed,and features for each candidate preposition are generated. Finally, supervised classification is applied, selecting the suggested preposition for every sentence.

Table 3: Sample PMI scores for the tokenization example.

N-gram	PMI scores by preposition
force be PREP	(with, -4.9), (under, -7.86), (at, -9.26), (in, -9.93), ...
be PREP you	(with, -1.86), (amongst, -1.99), (beside, -2.26), ...
PREP you .	(behind, -0.71), (beside, -0.82), (around, -0.84), ...

or a phrase in a sentence. While previous approaches considered only a short window of other n-grams in a certain vicinity near the preposition, we are instead considering all possible n-grams in a sentence. In order to limit the complexity and avoid considering a large number of n-grams, we start from the preposition position and build n-grams by limiting the n-gram distance (see Section 4.2) we consider. We experimentally compare different n-gram distances in Section 6.3.

In detail, for all n-grams composing a sentence, our system generates a ranking of prepositions according to some n-gram association measure. The n-gram association measure is used to compute a score that is proportional to the probability of an n-gram appearing together with a given preposition. In the literature, a number of association measures between words were proposed (see [14], Chapter 5), each one having its own advantages and disadvantages. In this work, we experimented with three association measures: *Point-wise Mutual Information* (PMI) [2], a variant of the Mutual Information and the Student's t-test[14], and found that PMI yields the best results in our context.

Briefly, PMI measures the gap between the probability of two variables being the same given their joint distribution and their individual distributions. The statistical frequencies we use to compute PMI-based rankings are taken from the *Google N-gram corpus* [15]. This corpus represents a collection of statistical n-gram data obtained from English books, and given its large size it helps to overcome PMI's inconsistent ranking on sparse data. Sample PMI scores for our running tokenization example are presented in Table 3.

4.4 N-grams with determiner skips

Determiners represent the most commonly used *part of speech* in English. We found that in both our training and test collections, around 30% of the prepositions are used in proximity to determiners such as "a" and "the". Generally speaking, determiners do not influence the choice of a particular preposition in a phrase. Figure 3 shows the distribution of correct prepositions ranks for two POS tags: determiners and nouns. We can confirm that the rank distribution for determiners is somewhat random, while for nouns there exists a clear bias towards higher ranks.

Given these observations, we generate so-called *skip n-grams* during the n-gram extraction process, where we strip determiners from the n-grams, whenever present. To better understand this process, let us consider the short phrase "one of the most". Without skip n-grams, we receive the following trigrams during extraction: "one of the", "of the most". When we apply determiner skips, we produce an additional trigram: "one of most".

Since the Google N-gram corpus does not contain skip n-grams directly, we produce them ourselves to get the correct counts for all n-grams containing any of the possible determiners[9].

4.5 Preposition selection

Given the generated ranking of prepositions for each individual n-gram, our objective is to select the right preposition in a sentence. To achieve this goal, we apply a supervised learning method by means of two-class classification.

For every preposition occurrence in the text, we generate a number of potential replacements equal to the size of our candidate preposition set. Each potential replacement receives its own feature values that correspond to a particular preposition from the candidate set. These feature values incorporate the individual n-gram rankings we introduced earlier, and the classifier makes a binary decision on whether or not a particular preposition is correct for a given sentence. Each decision is given with an accompanying confidence score.[10] In the following section, we discuss the set of features we use in our supervised classification step.

[9]We use the following list of determiners: a, an, the, my, our, your, their.

[10]Generally speaking, more than one preposition can be classified as correct using this classification approach. In such

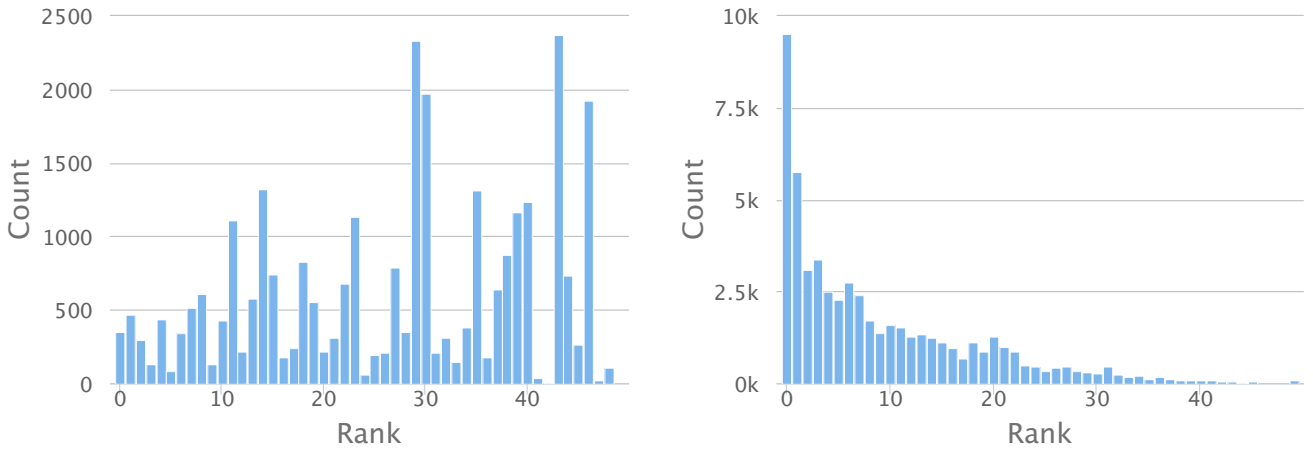

Figure 3: Distribution of correct prepositions ranks for determiners (left) and nouns (right). The statistics are based on the FCE collection.

5. FEATURES FOR PREPOSITION SELECTION

5.1 PMI-based features

As outlined in Section 4.3, we use PMI scores to obtain a ranking of prepositions for every n-gram. A possible way to select prepositions is to use the numerical rank of a preposition directly as a measure of association.

The PMI score has the following property: A positive PMI score signals the presence of some association, while a negative score indicates disassociation. Thus, we can directly use the PMI score of a preposition as a feature for a supervised approach.

Based on these observations, we propose the following set of features:

- Average rank of a preposition among the preposition ranks of the considered n-grams;
- Average PMI score of a preposition among the PMI scores of the considered n-grams;
- Total number of occurrences of a certain preposition on the first place of the ranking among the ranks of the considered n-grams.

For each of the three features listed above, it is possible to calculate the various scores of different logical groups of n-grams; In this work, we group by n-gram size (unigram, bigram, etc.) and by n-gram distances. Thus, in total we get $3 * n + 3 * k$ number of features, where n is the number of different n-gram sizes and k is the number of different n-gram distances.

5.2 Central n-grams

The n-grams that are central to a preposition (distance=0) stand aside other n-grams, since they contain both left and right contexts for a given preposition. Figure 4 shows how often a top-ranked preposition for an n-gram is correct with respect to the n-gram distance.

We observe that central n-grams represent the largest chunk in the distribution. Therefore, it is important to incorporate their preposition rankings as separate features.

cases, we select the most likely preposition according to the classifier's confidence score.

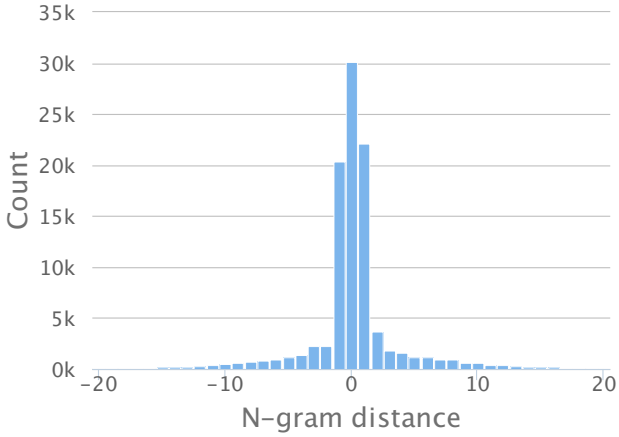

Figure 4: Distribution of correct preposition counts on top of PMI ranks with respect to n-gram distance. The statistics are based on the FCE collection.

Following the features designed in Section 5.1, we add the PMI score and the rank of a given preposition from a central n-gram as features. Since there might be no central n-gram for some sentences, we also add a binary indicator feature to indicate whether or not such an n-gram is present. In case both skip and non-skip central n-grams coexist, we use the skip n-grams.

5.3 Features based on confusion matrix

The selection of the correct preposition can also be made based on the currently observed preposition. Non-native English speakers tend to commonly substitute a correct preposition using the same, incorrect preposition, depending on the learner's skills and background. In this case, the observed (but wrong) preposition directly correlates to a correct one, and we use this as a potentially highly discriminative feature.

Since we are generating potential replacements for every possible preposition, we can directly leverage the probability of the observed preposition as learned from the probability

matrix of the training collection (Table 4), which indicates the probability of a given preposition being used instead of another preposition.

Another valuable information in the matrix is that some correct (and observed) preposition pairs have very low or zero probability: This represents valuable evidence indicating that certain substitutions are very unlikely to happen and should not be considered by our correction approach.

Thus, for each preposition from our candidate set, we take its probability from the confusion matrix given an original preposition and use it as a feature (See, for instance, Table 4).

5.4 Part-of-Speech Tags

Part-of-Speech (POS) tags have often been considered as an important discriminative feature for many NLP tasks. Many previous contributions on grammatical error correction used POS tag information from the context words surrounding the preposition and proved it to be important. In this work, we also take the top-5 most frequent POS tags that either immediately follow or precede a preposition and use them as binary features in our classifier (see Figure 5). The n-grams whose POS tags do no match the top-5 tags are assigned to the category "OTHER", which is also used as a binary feature.

5.5 Other features

In addition to the features described above, we also add binary features that represent the preposition itself. That is, we generate a sparse vector of size equal to our candidate preposition set with only one value equal to one, denoting the currently observed preposition.

6. EXPERIMENTAL EVALUATION

As outlined in Section 4.5, given a sentence with a preposition, our goal is to rank every preposition from the candidate set according to their likelihood of being the correct one based on the context (n-grams) surrounding the preposition. In this paper, we consider n-grams with $n \in \{2, 3\}$ including the preposition itself. For skip n-grams (see Section 4.4), we also consider $n = 4$, including prepositions and determiners.

The classification itself was performed using a random forest classifier [7] using an implementation from the scikit-learn package [17]. This type of classifier allows to automatically rank features by their importance score without the need to evaluate different feature combinations manually.

As we can see from Table 1, preposition errors account for just 5% of all preposition occurrences in both training and CoNLL-2013 test collections. Furthermore, the features based on the confusion matrix highly correlate the observed prepositions with the correct ones. Thus, we need to balance the training collection with a similar amount of negative and positive examples.

By experimentally comparing different balancing methods on cross-validated training collections, we found that our classifier performs best when under-sampling non-errors so that the amount of real error samples is equal to the amount of non-errors. Both oversampling and under-sampling were performed using uniform random sampling [11].

[11]We also tried to sample while keeping the proportion of prepositions equal to the ones in the original collection, but did not observe any improvement over random sampling.

Table 5: Features ranked by their importance scores computed over the training collection.

Feature name		Importance score
conf_matrix_score	(Sec. 5.3)	0.28
top_prep_count_2gram	(Sec. 5.1)	0.13
avg_rank_dist0	(Sec. 5.1)	0.06
central_ngram_rank	(Sec. 5.2)	0.06
avg_rank_dist1	(Sec. 5.1)	0.05

The rest of this section is structured as follows: First, we discuss our approach to hyper-parameter optimization to prevent over-fitting the training data and show which features play the most important role for the classifier. We then evaluate the effects of restricting the distance of n-grams on the classifier's performance. Next, we analyze the impact of restricting the size of the n-grams as well as the impact of skip n-grams. Finally, we perform an evaluation of our classifier on the standard CoNLL-2013 collection by comparing our approach against the top-performing classifier from the CoNLL-2013 Shared Task, as well as on the SE test collection we built.

6.1 Hyper-Parameter Optimization

In any classifier based on decision trees, it is possible to optimize at least two parameters: the depth of the tree, and the minimum number of samples in a leaf. By restricting both of parameters to a certain range, we can effectively prevent the classifier from over-fitting the training data. Figure 6 shows the F1 values resulting from different combinations of depths and minimum samples using 10-fold cross-validation on the training collection. We observe that the best results are achieved with deeper trees and a small number of minimum samples in a leaf. Given these results, we fix the range of possible depth values between 5 and 50, and the range of minimum number of samples between 1 and 100. The results presented in Sections 6.2, 6.3 and 6.4 report maximum scores by taking all possible combinations of the hyper-parameters into account. Each score itself represents an average score obtained through 10-fold cross-validation.

6.2 Feature importance

Table 5 shows the top-5 features ranked by the importance scores assigned by the classifier. As expected, we observe that the feature based on the confusion matrix score is selected as most significant by the classifier. This is explained by the fact that in ∼95% of the cases there is no need to change the original preposition. The second most important feature is the total count of the preposition appearing on the first place of the n-gram rankings based on PMI. The next most relevant features include the preposition rank of the central n-gram and the average ranks grouped by n-gram distance.

6.3 N-Gram Distance Restriction

According to Section 5.2, the number of n-grams with a correct preposition on top of the ranking decreases with the increase of the absolute distance to a preposition. By considering the n-grams distances from a preposition, we can evaluate if restricting the distance affects the results

Table 4: Preposition probability matrix for the 10 most frequent prepositions we consider. In the columns we represent the original prepositions found in the sentences while in the rows we represent the correct prepositions. Thus, a matrix entry indicates the probability that the preposition in the column has to be replaced with the preposition in the row.

Target Prep.	Original Preposition									
	to	**in**	**of**	**for**	**on**	**but**	**at**	**with**	**about**	**from**
to	0.958	0.007	0.002	0.011	0.004	0	0.003	0.005	0	0.002
in	0.037	0.79	0.01	0.009	0.066	0	0.036	0.015	0.001	0.008
of	0.013	0.017	0.894	0.024	0.012	0	0.003	0.005	0.007	0.008
for	0.065	0.012	0.01	0.865	0.013	0.0	0.002	0.009	0.01	0.006
on	0.037	0.182	0.02	0.012	0.669	0	0.026	0.017	0.006	0.006
but	0	0	0	0	0	0.992	0	0.001	0	0
at	0.02	0.051	0.003	0.004	0.022	0	0.885	0.005	0	0.003
with	0.039	0.023	0.01	0.009	0.01	0	0.005	0.868	0.006	0.007
about	0.005	0.005	0.018	0.021	0.006	0	0.001	0.021	0.916	0.002
from	0.008	0.023	0.022	0.018	0.011	0	0.003	0.011	0.006	0.875

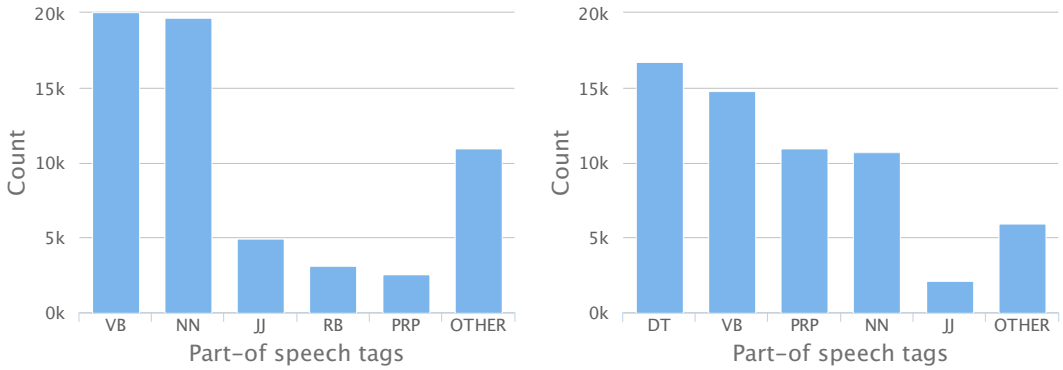

Figure 5: Top 5 most frequent part-of-speech tags from the training collection that precede (left) or follow (right) preposition.

and which combinations of distances yield the best results. Another benefit of restricting the distance is efficiency: the less n-grams we need to consider, the faster we can compute PMI scores.

To find the optimal set of distances, we compare the performance of our classifier by restricting the distances to certain ranges. The results of this comparison are shown in Table 6. The best performing approach restricts the distance in the $(-2, 2)$ range, but it is not significantly different from the $(-1, 1)$ range. Therefore, we decided to use n-grams in the $(-1, 1)$ distance range for all further experiments, which drastically reduces the number of n-grams to consider, thus making our approach much more efficient.

Table 6: Effectiveness values for different n-gram distance restrictions. The symbol * indicates a statistically significant difference (t-test $p < 0.01$) as compared to the best performing approach (bold).

Distance Restriction	Precision	Recall	F1 score
(0)	0.3077*	0.3908*	0.3442*
$(-1, 1)$	**0.3231**	0.4166	0.3637
$(-2, 2)$	0.3214	**0.4222**	**0.3648**
$(-5, 5)$	0.3223	0.4028	0.3577
No restriction	0.3214	0.3924*	0.3532

6.4 N-gram Sizes

In the following, we compare the effectiveness of our system by restricting the set of permitted n-gram sizes. The results of this experiment are presented in Table 7. We can confirm that the introduction of skip n-grams (i.e., results using $n = 4$) contributes to better final classification results. We also observe that the unrestricted set of n-gram sizes yields the best score.

6.5 CoNLL-2013 Collection Evaluation

Finally, we evaluate our classifier on two different test collections. At first, we evaluate our approach using the test collection of the CoNLL-2013 Shared Task. For this experiment, we trained the classifier on the complete training collection and set the other parameters to the ones that yield the best scores according to the previous experiments run on the Cambridge FCE collection. Table 8 shows the results of

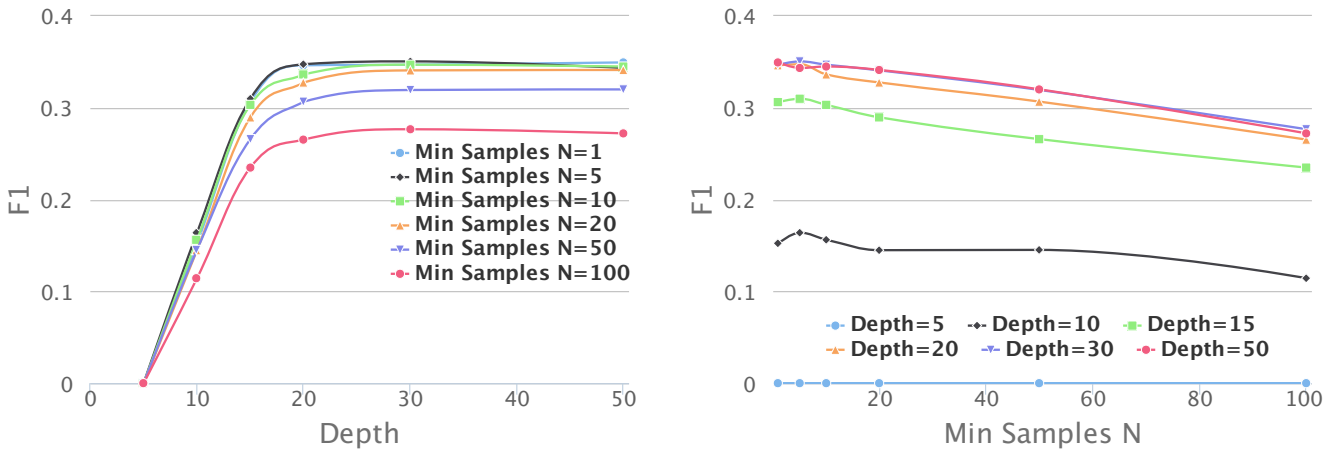

Figure 6: F1 scores for different combinations of hyper-parameters.

Table 8: Effectiveness values for different approaches on the CoNLL-2013 and SE test collections. Results for our approach are reported with 95% confidence intervals based on standard deviation.

Collection	Approach	Precision	Recall	F1 score
CoNLL-2013	NARA Team @CoNLL2013	**0.2910**	0.1254	0.1753
	N-gram-based classification	0.2592 ± 0.010	**0.3611±0.010**	**0.3017±0.0082**
SE	N-gram-based classification	0.1585 ± 0.0145	0.2185 ± 0.021	0.1837 ± 0.0171
	N-gram-based classification (cross-validation)	0.2704 ± 0.0283	0.2961 ± 0.0362	0.2824 ± 0.0313

Table 7: Effectiveness values for different combinations of n-gram sizes. The symbol * indicates a statistically significant difference (t-test $p < 0.01$) as compared to the best performing approach (bold).

N-gram	Precision	Recall	F1 score
$\{2,3\}$-grams	0.3005^*	0.3879^*	0.3385^*
$\{3,4\}$-grams	0.2931^*	**0.4187**	0.3447^*
$\{2,3,4\}$-grams	**0.3231**	0.4166	**0.3637**

this evaluation. We observe that our classifier clearly outperforms the best approach from the CoNLL-2013 Shared Task, with a 13% absolute and 76% relative improvement.

6.6 SE Collection Evaluation

In the second part of the evaluation, we consider the Stack Exchange test collection (see Section 3.2). For this collection, we evaluate the effectiveness of our classifier in two different setups. In the first experiment, we simply take the trained classifier used for the CoNLL-2013 collection, and apply it on the SE collection. Preposition errors are randomly under-sampled so that they constitute 5% of the training set. The experiment is repeated 10 times and we report the mean value in Table 8. In the second experiment, we perform 10-fold cross-validation on the SE collection, where the test part is sampled similarly as for the first experiment.

We can see that when evaluating our approach by cross-validation on a collection built on top of user-generated content, we obtain effectiveness scores (i.e., 28% F1) comparable to the ones obtained on the CoNLL-2013 collection indicating the robustness of the proposed approach. While training our classifier on the CoNLL-2013 collection and applying it on the SE collection yields a drop in performance, our approach still obtains reasonable results indicating its portability to different domains.

7. CONCLUSIONS

Identifying and correcting grammatical errors in textual content is important for many applications. In this paper, we focused on the correction of preposition errors in English text. We proposed a supervised approach that uses a set of features designed around the notion of n-gram rankings. Our system uses a large collection of English books as evidence of correct preposition usage and generates a ranking of candidate prepositions for replacement in the sentence. The decision of which preposition to use is then made at the sentence-level, through a binary-classification step. We evaluated our techniques over a standard evaluation collection as well as over a newly created collection of UGC. We confirmed with extensive experiments that n-gram-based classification and preposition ranking outperforms more complex multi-class classification methods for the task of preposition error correction.

As future work, we would like to address the issue of incomplete n-gram counts. For example, when a certain n-gram represents a very specific concept or a new *entity*, we might not find enough evidence to understand its correct usage in our n-gram corpus. However, it could be possible to use entity type information (e.g., Actor) to find similar

entities and use their counts instead. Finally, as our preposition classification approach allows to select multiple valid prepositions for a given sentence, we plan to broaden our approach and evaluation methodology to determine when a sentence indeed permits multiple valid preposition choices.

8. ACKNOWLEDGMENT

This work was supported by the Swiss National Science Foundation under grant numbers PP00P2_128459 and 200021_143649.

9. REFERENCES

[1] S. Bergsma, D. Lin, and R. Goebel. Web-scale n-gram models for lexical disambiguation. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, IJCAI'09, pages 1507–1512, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[2] K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29, Mar. 1990.

[3] D. Dahlmeier and H. T. Ng. A beam-search decoder for grammatical error correction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 568–578, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[4] D. Dahlmeier, H. T. Ng, and E. J. F. Ng. Nus at the hoo 2012 shared task. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 216–224, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[5] R. Dale, I. Anisimoff, and G. Narroway. Hoo 2012: A report on the preposition and determiner error correction shared task. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 54–62, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[6] A. Elghafari, D. Meurers, and H. Wunsch. Exploring the data-driven prediction of prepositions in english. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 267–275, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[7] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 63(1):3–42, Apr. 2006.

[8] M. Heilman, A. Cahill, and J. Tetreault. Precision isn't everything: A hybrid approach to grammatical error detection. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 233–241, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[9] J. Huang, J. Gao, J. Miao, X. Li, K. Wang, F. Behr, and C. L. Giles. Exploring web scale language models for search query processing. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 451–460, New York, NY, USA, 2010. ACM.

[10] A. Islam and D. Inkpen. An unsupervised approach to preposition error correction. In *Natural Language Processing and Knowledge Engineering (NLP-KE), 2010 International Conference on*, pages 1–4. IEEE, 2010.

[11] T.-h. Kao, Y.-w. Chang, H.-w. Chiu, T.-H. Yen, J. Boisson, J.-c. Wu, and J. S. Chang. Conll-2013 shared task: Grammatical error correction nthu system description. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 20–25, Sofia, Bulgaria, 2013. Association for Computational Linguistics.

[12] E. Kochmar, O. Andersen, and T. Briscoe. Hoo 2012 error recognition and correction shared task: Cambridge university submission report. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 242–250, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[13] C. Leacock, M. Chodorow, M. Gamon, and J. Tetreault. *Automated Grammatical Error Detection for Language Learners*. Morgan and Claypool Publishers, 2010.

[14] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[15] J.-B. Michel, Y. K. Shen, A. P. Aiden, A. Veres, M. K. Gray, T. G. B. Team, J. P. Pickett, D. Hoiberg, D. Clancy, P. Norvig, J. Orwant, S. Pinker, M. A. Nowak, and E. L. Aiden. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182, 2011.

[16] H. T. Ng, S. M. Wu, Y. Wu, C. Hadiwinoto, and J. Tetreault. The conll-2013 shared task on grammatical error correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–12, Sofia, Bulgaria, 2013. Association for Computational Linguistics.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[18] R. Prokofyev, G. Demartini, and P. Cudré-Mauroux. Effective named entity recognition for idiosyncratic web collections. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 397–408, Republic and Canton of Geneva, Switzerland, 2014. International World Wide Web Conferences Steering Committee.

[19] A. Rozovskaya, K.-W. Chang, M. Sammons, and D. Roth. The university of illinois system in the conll-2013 shared task. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 13–19, Sofia, Bulgaria, 2013. Association for Computational Linguistics.

[20] Y. Xiang, B. Yuan, Y. Zhang, X. Wang, W. Zheng, and C. Wei. A hybrid model for grammatical error correction. *CoNLL-2013*, page 115, 2013.