Generating artificial errors for grammatical error correction

Mariano Felice
Computer Laboratory
University of Cambridge
United Kingdom
mf501@cam.ac.uk

Zheng Yuan
Computer Laboratory
University of Cambridge
United Kingdom
zy249@cam.ac.uk

Abstract

This paper explores the generation of artificial errors for correcting grammatical mistakes made by learners of English as a second language. Artificial errors are injected into a set of error-free sentences in a probabilistic manner using statistics from a corpus. Unlike previous approaches, we use linguistic information to derive error generation probabilities and build corpora to correct several error types, including open-class errors. In addition, we also analyse the variables involved in the selection of candidate sentences. Experiments using the NUCLE corpus from the CoNLL 2013 shared task reveal that: 1) training on artificially created errors improves precision at the expense of recall and 2) different types of linguistic information are better suited for correcting different error types.

1 Introduction

Building error correction systems using machine learning techniques can require a considerable amount of annotated data which is difficult to obtain. Available error-annotated corpora are often focused on particular groups of people (e.g. non-native students), error types (e.g. spelling, syntax), genres (e.g. university essays, letters) or topics so it is not clear how representative they are or how well systems based on them will generalise. On the other hand, building new corpora is not always a viable solution since error annotation is expensive. As a result, researchers have tried to overcome these limitations either by compiling corpora automatically from the web (Mizumoto et al., 2011; Tajiri et al., 2012; Cahill et al., 2013) or using artificial corpora which are cheaper to produce and can be tailored to their needs.

Artificial error generation allows researchers to create very large error-annotated corpora with little effort and control variables such as topic and error types. Errors can be injected into candidate texts using a deterministic approach (e.g. fixed rules) or probabilities derived from manually annotated samples in order to mimic real data.

Although artificial errors have been used in previous work, we present a new approach based on linguistic information and evaluate it using the test data provided for the CoNLL 2013 shared task on grammatical error correction (Ng et al., 2013).

Our work makes the following contributions. First, we are the first to use linguistic information (such as part-of-speech (PoS) information or semantic classes) to characterise contexts of naturally occurring errors and replicate them in error-free text. Second, we apply our technique to a larger number of error types than any other previous approach, including open-class errors. The resulting datasets are used to train error correction systems aimed at learners of English as a second language (ESL). Finally, we provide a detailed description of the variables that affect artificial error generation.

2 Related work

The use of artificial data to train error correction systems has been explored by other researchers using a variety of techniques.

Izumi et al. (2003), for example, use artificial errors to target article mistakes made by Japanese learners of English. A corpus is created by replacing *a*, *an*, *the* or the zero article by a different article chosen at random in more than 7,500 correct sentences and used to train a maximum entropy model. Results show an improvement for omission errors but no change for replacement errors.

Brockett et al. (2006) describe the use of a statistical machine translation (SMT) system for correcting a set of 14 countable/uncountable nouns

Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 116–126,
Gothenburg, Sweden, April 26-30 2014.

which are often confusing for ESL learners. Their training corpus consists of a large number of sentences extracted from news articles which were deliberately modified to include typical countability errors based on evidence from a Chinese learner corpus. Their approach to artificial error injection is deterministic, using hand-coded rules to change quantifiers (*much* → *many*), generate plurals (*advice* → *advices*) or insert unnecessary determiners. Experiments show their system was generally able to beat the standard Microsoft Word 2003 grammar checker, although it produced a relatively higher rate of erroneous corrections.

SMT systems are also used by Ehsan and Faili (2013) to correct grammatical errors and context-sensitive spelling mistakes in English and Farsi. Training corpora are obtained by injecting artificial errors into well-formed treebank sentences using predefined error templates. Whenever an original sentence from the corpus matches one of these templates, a pair of correct and incorrect sentences is generated. This process is repeated multiple times if a single sentence matches more than one error template, thereby generating many pairs for the same original sentence. A comparison between the proposed systems and rule-based grammar checkers show they are complementary, with a hybrid system achieving the best performance.

2.1 Probabilistic approaches

A few researchers have explored probabilistic methods in an attempt to mimic real data more accurately. Foster and Andersen (2009), for example, describe a tool for generating artificial errors based on statistics from other corpora, such as the Cambridge Learner Corpus (CLC).[1] Their experiments show a drop in accuracy when artificial sentences are used as a replacement for real incorrect sentences, suggesting that they may not be as useful as genuine text. Their report also includes an extensive summary of previous work in the area.

Rozovskaya and Roth propose more sophisticated probabilistic methods to generate artificial errors for articles (2010a) and prepositions (2010b; 2011), also based on statistics from an ESL corpus. In particular, they compile a set of sentences from the English Wikipedia and apply the following generation methods:

General

Target words (e.g. articles) are replaced with others of the same class with probability x (varying from 0.05 to 0.18). Each new word is chosen uniformly at random.

Distribution before correction (in ESL data)

Target words in the error-free text are changed to match the distribution observed in ESL error-annotated data before any correction is made.

Distribution after correction (in ESL data)

Target words in the error-free text are changed to match the distribution observed in ESL error-annotated data after corrections are made.

Native language-specific distributions

It has been observed that second language production is affected by a learner's native language (L1) (Lee and Seneff, 2008; Leacock et al., 2010). A common example is the difficulty in using English articles appropriately by learners whose L1 has no article system, such as Russian or Japanese. Because word choice errors follow systematic patterns (i.e. they do not occur randomly), this information is extremely valuable for generating errors more accurately.

L1-specific errors can be imitated by computing word confusions in an error-annotated ESL corpora and using these distributions to change target words accordingly in error-free text. More specifically, if we estimate P(source|target) in an error-tagged corpus (i.e. the probability of an incorrect *source* word being used when the correct *target* is expected), we can generate more accurate confusion sets where each candidate has an associated probability depending on the observed word. For example, supposing that a group of learners use the preposition *to* in 10% of cases where the preposition *for* should be used (that is, P(source=*to*|target=*for*)=0.10), we can replicate this error pattern by replacing the occurrences of the preposition *for* with *to* with a probability of 0.10 in a corpus of error-free sentences. When the source and target words are the same, P(source=*x*|target=*x*) expresses the probability that a learner produces the correct/expected word.

Because errors are generally sparse (and therefore error rates are low), replicating mistakes based on observed probabilities can easily lead to

[1] http://www.cup.cam.ac.uk/gb/elt/catalogue/subject/custom/item3646603/Cambridge-International-Corpus-Cambridge-Learner-Corpus/

low recall. In order to address this issue during artificial error generation, Rozovskaya et al. (2012) propose an *inflation method* that boosts confusion probabilities in order to generate a larger proportion of artificial instances. This reformulation is shown to improve F-scores when correcting determiners and prepositions.

Experiments reveal that these approaches yield better results than assuming uniform probabilistic distributions where all errors and corrections are equally likely. In particular, classifiers trained on artificially generated data outperformed those trained on native error-free text (Rozovskaya and Roth, 2010a; Rozovskaya and Roth, 2011). However, it has also been shown that using artificially generated data as a replacement for non-native error-corrected data can lead to poorer performance (Sjöbergh and Knutsson, 2005; Foster and Andersen, 2009). This would suggest that artificial errors are more useful than native data but less useful than corrected non-native data.

Rozovskaya and Roth also control other variables in their experiments. On the one hand, they only evaluate their systems on sentences that have no spelling mistakes so as to avoid degrading performance. This is particularly important when training classifiers on features extracted with linguistic tools (such as parsers or taggers) as they could provide inaccurate results for malformed input. On the other hand, the authors work on a limited set of error types (mainly articles and prepositions) which are closed word classes and therefore have reduced confusion sets. Thus, it becomes interesting to investigate how their ideas extrapolate to open-class error types, like verb form or content word errors.

Their probabilistic error generation approach has also been used by other researchers. Imamura et al. (2012), for example, applied this method to generate artificial incorrect sentences for Japanese particle correction with an inflation factor ranging from 0.0 (no errors) to 2.0 (double error rates). Their results show that the performance of artificial corpora depends largely on the inflation rate but can achieve good results when domain adaptation is applied.

In a more exhaustive study, Cahill et al. (2013) investigate the usefulness of automatically-compiled sentences from Wikipedia revisions for correcting preposition errors. A number of classifiers are trained using error-free text, automatically-compiled annotated corpora and artificial sentences generated using error probabilities derived from Wikipedia revisions and Lang-8.[2] Their results reveal a number of interesting points, namely that artificial errors provide competitive results and perform robustly across different test sets. A learning curve analysis also shows system performance increases as more training data is used, both real and artificial.

More recently, some teams have also reported improvements by using artificial data in their submissions to the CoNLL 2013 shared task. Rozovskaya et al. (2013) apply their inflation method to train a classifier for determiner errors that achieves state-of-the-art performance while Yuan and Felice (2013) use naively-generated artificial errors within an SMT framework that places them third in terms of precision.

3 Advanced generation of artificial errors

Our work is based on the hypothesis that using carefully generated artificial errors improves the performance of error correction systems. This implies generating errors in a way that resembles available error-annotated data, using similar texts and accurate injection methods. Like other probabilistic approaches, our method assumes we have access to an error-corrected reference corpus from which we can compute error generation probabilities.

3.1 Base text selection

We analyse a set of variables that we consider important for collecting suitable texts for error injection, namely:

Topic

Replicating errors on texts about the same topic as the training/test data is more likely to produce better results than out-of-domain data, as vocabulary and word senses are more likely to be similar. In addition, similar texts are more likely to exhibit suitable contexts for error injection and consequently help the system focus on particularly useful information.

Genre

In cases where no a priori information about topic is available (for example, because the test set is

[2] http://lang-8.com/

unknown or the system will be used in different scenarios), knowing the genre or type of text the system will process can also be useful. Example genres include expository (descriptions, essays, reports, etc.), narrative (stories), persuasive (reviews, advertisements, etc.), procedural (instructions, recipes, experiments, etc.) and transactional texts (letters, interviews, etc.).

Style/register

As with the previous aspects, style (colloquial, academic, etc.) and register (from formal written to informal spoken) also affect production and should therefore be modelled accurately in the training data.

Text complexity/language proficiency

Candidate texts should exhibit the same reading complexity as training/test texts and be written by or targeted at learners with similar English proficiency. Otherwise, the overlap in vocabulary and grammatical structures is more likely to be small and thus hinder error injection.

Native language

Because second language production is known to be affected by a learner's L1, using candidate texts produced by groups of the same L1 as the training/test data should provide more suitable contexts for error injection. When such texts are not available, using data by speakers of other L1s that exhibit similar phenomena (e.g. no article system, agglutinative languages, etc.) might also be useful. However, finding error-free texts written in English by a specific population can be difficult, which is why most approaches resort to native English text.

In our experiments, the aforementioned variables are manually controlled although we believe many of them could be assessed automatically. For example, topics could be estimated using text similarity measures, genres could be predicted using structural information and L1s could be inferred using a native language identifier.[3]

For an analysis of other variables such as domain and error distributions, the reader should refer to Cahill et al. (2013).

[3]See the First Edition of the Shared Task on Native Language Identification (Tetreault et al., 2013) at `https://sites.google.com/site/nlisharedtask2013/`

3.2 Error replication

Our approach to artificial error generation is similar to the one proposed by Rozovskaya and Roth (2010a) in that we also estimate probabilities in a corpus of ESL learners which are then used to distort error-free text. However, unlike them, we refine our probabilities by imposing restrictions on the linguistic functions of the words and the contexts where they occur. Because we extend generation to open-class error types (such as verb form errors), this refinement becomes necessary to overcome disambiguation issues and lead to more accurate replication.

Our work is the first to exploit linguistic information for error generation, as described below.

Error type distributions

We compute the probability of each error type $p(t)$ occurring over the total number of relevant instances (e.g. noun phrases are relevant instances for article errors). During generation, $p(t)$ is uniformly distributed over all the possible choices for the error type (e.g. for articles, choices are *a*, *an*, *the* or the zero article). Relevant instances are detected in the base text and changed for an alternative at random using the estimated probabilities. The probability of leaving relevant instances unchanged is $1 - p(t)$.

Morphology

We believe morphological information such as person or number is particularly useful for identifying and correcting specific error types, such as articles, noun number or subject-verb agreement. Thus, we compute the conditional probability of words in specific classes for different morphological contexts (such as noun number or PoS). The following example shows confusion probabilities for singular head nouns requiring *an*:

P(source-det=*an*|target-det=$an_{\text{head-noun=NN}}$) = 0.942
P(source-det=*the*|target-det=$an_{\text{head-noun=NN}}$) = 0.034
P(source-det=*a*|target-det=$an_{\text{head-noun=NN}}$) = 0.015
P(source-det=other|target-det=$an_{\text{head-noun=NN}}$) = 0.005
P(source-det=$\varnothing$|target-det=$an_{\text{head-noun=NN}}$) = 0.004

PoS disambiguation

Most approaches to artificial error generation are aimed at correcting closed-class words such as articles or prepositions, which rarely occur with

a different part of speech in the text. However, when we consider open-class error types, we should perform PoS disambiguation since the same surface form could play different roles in a sentence. For example, consider generating artificial verb form errors for the verb *to play* after observing its distribution in an error-annotated corpus. By using PoS tags, we can easily determine if an occurrence of the word *play* is a verb or a noun and thus compute or apply the appropriate probabilities:

P(source=*play*|target=$play_{\mathrm{V}}$) = 0.98
P(source=*plays*|target=$play_{\mathrm{V}}$) = 0.02

P(source=*play*|target=$play_{\mathrm{N}}$) = 0.84
P(source=*plays*|target=$play_{\mathrm{N}}$) = 0.16

Semantic classes

We hypothesise that semantic information about concepts in the sentences can shed light on specific usage patterns that may otherwise be hidden. For example, we could refine confusion sets for prepositions according to the type of object they are applied to (a location, a recipient, an instrument, etc.):

P(prep=*in*|noun_class=location) = 0.39
P(prep=*to*|noun_class=location) = 0.31
P(prep=*at*|noun_class=location) = 0.16
P(prep=*from*|noun_class=location) = 0.07
P(prep=∅|noun_class=location) = 0.05
P(prep=other|noun_class=location) = 0.03

By abstracting from surface forms, we can also generate faithful errors for words that have not been previously observed, e.g. we may have not seen *hospital* but we may have seen *school*, *my sister's house* or *church*.

Word senses

Polysemous words with the same PoS can exhibit different patterns of usage for each of their meanings (e.g. one meaning may co-occur with a specific preposition more often than the others). For this reason, we introduce probabilities for each word sense in an attempt to capture more accurate usage. As an example, consider a hypothetical situation in which a group of learners confuse prepositions used with the word *bank* as a financial institution but they produce the right preposition when it refers to a river bed:

P(prep=*in*|noun=bank_1) = 0.76
P(prep=*at*|noun=bank_1) = 0.18
P(prep=*on*|noun=bank_1) = 0.06

P(prep=*on*|noun=bank_2) = 1.00

Although it is rare that occurrences of the same word will refer to different meanings within a document (the so-called 'one sense per discourse' assumption (Gale et al., 1992)), this is not the case when large corpora containing different documents are used for characterising and generating errors. In such scenarios, word sense disambiguation should produce more accurate results.

Table 1 lists the actual probabilities computed from each type of information and the errors they are able to generate.

4 Experimental setup

4.1 Corpora and tools

We use the NUCLE v2.3 corpus (Dahlmeier et al., 2013) released for the CoNLL 2013 shared task on error correction, which comprises error-annotated essays written in English by students at the National University of Singapore. These essays cover topics such as environmental pollution, health care, welfare, technology, etc. All the sentences were manually annotated by human experts using a set of 27 error types, but we used the filtered version containing only the five types selected for the shared task: ArtOrDet (article or determiner), Nn (noun number), Prep (preposition), SVA (subject-verb agreements) and Vform (verb form) errors. The training set of the NUCLE corpus contains 57,151 sentences and 1,161,567 tokens while the test set comprises 1,381 sentences and 29,207 tokens. The training portion of the corpus was used to estimate the required conditional probabilities and train a few variations of our systems while the test set was reserved to evaluate performance.

Candidate native texts for error injection were extracted from the English Wikipedia, controlling the variables described Section 3.1 as follows:

Topic: We chose an initial set of 50 Wikipedia articles based on keywords in the NUCLE training data and proceeded to collect related articles by following hyperlinks in their 'See also' section. We retrieved a total of 494 articles which were later preprocessed to remove

Information	Probability	Generated error types
Error type distribution	P(*error_type*)	ArtOrDet, Nn, Prep, SVA, Vform
Morphology	P(source=*determiner*|target=*determiner*, *head_noun_tag*) P(source=*verb_tag*|target=*verb_tag*, *subj_head_noun_tag*)	ArtOrDet, SVA
PoS disambiguation	P(source=*word*|target=*word*, *PoS*)	Nn, Vform
Semantic classes	P(source=*determiner*|target=*determiner*, *head_noun_class*) P(source=*preposition*|target=*preposition*, *head_noun_class*)	ArtOrDet, Prep
Word senses	P(source=*preposition*|*verb_sense* + *obj_head_noun_sense*) P(source=*preposition*|target=*preposition*, *head_noun_sense*) P(source=*preposition*|target=*preposition*, *dep_adj_sense*) P(source=*determiner*|target=*determiner*, *head_noun_sense*) P(source=*verb_tag*|target=*verb_tag*, *subj_head_noun_sense*)	ArtOrDet, Prep, SVA

Table 1: Probabilities computed for each type of linguistic information. Error codes correspond to the five error types in the CoNLL 2013 shared task: ArtOrDet (article or determiner), Nn (noun number), Prep (prepositions), SVA (subject-verb agreement) and Vform (verb form).

wikicode tags, yielding 54,945 sentences and approximately 1,123,739 tokens.

Genre: Both NUCLE and Wikipedia contain expository texts, although they are not necessarily similar.

Style/register: Written, academic and formal.

Text complexity/language proficiency: Essays in the NUCLE corpus are written by advanced university students and are therefore comparable to standard English Wikipedia articles. For less sophisticated language, the Simple English Wikipedia could be an alternative.

Native language: English Wikipedia articles are mostly written by native speakers whereas NUCLE essays are not. This is the only discordant variable.

PoS tagging was performed using RASP (Briscoe et al., 2006). Word sense disambiguation was carried out using the WordNet::SenseRelate:AllWords Perl module (Pedersen and Kolhatkar, 2009) which assigns a sense from WordNet (Miller, 1995) to each content word in a text. As for semantic information, we use WordNet classes which are readily available in NLTK (Bird et al., 2009). WordNet classes respond to a classification in lexicographers' files[4] and are defined for content words as shown in Table 2, depending on their location in the hierarchy.

[4] http://wordnet.princeton.edu/man/lexnames.5WN.html

Part of speech	WordNet classification
Adjective	all, pertainyms, participial
Adverb	all
Noun	act, animal, artifact, attribute, body, cognition, communication, event, feeling, food, group, location, motive, object, person, phenomenon, plant, possession, process, quantity, relation, shape, state, substance, time
Verb	body, change, cognition, communication, competition, consumption, contact, creation, emotion, motion, perception, possession, social, stative, weather

Table 2: WordNet classes for content words.

Name	Composition
ED	errors based on error type distributions
MORPH	errors based on morphology
POS	errors based on PoS disambiguation
SC	errors based on semantic classes
WSD	errors based on word senses

Table 3: Generated artificial corpora based on different types of linguistic information.

4.2 Error generation

For each type of information in Table 1, we compute the corresponding conditional probabilities using the NUCLE training set. These probabilities are then used to generate six different artificial corpora using the *inflation method* (Rozovskaya et al., 2012), as listed in Table 3.

4.3 System training

We approach the error correction task as a translation problem from incorrect into correct English. Systems are built using an SMT framework and different combinations of NUCLE and our artificial corpora, where the source side contains in-

	Original						Revised					
	C	**M**	**U**	**P**	**R**	**F_1**	**C**	**M**	**U**	**P**	**R**	**F_1**
NUCLE (baseline)	181	1462	513	0.2608	0.1102	0.1549	200	1483	495	0.2878	0.1188	0.1682
ED	53	1590	150	**0.2611**	0.0323	0.0574	62	1621	141	**0.3054**	0.0368	0.0657
MORPH	74	1569	333	0.1818	0.0450	0.0722	83	1600	324	0.2039	0.0493	0.0794
POS	42	1601	99	**0.2979**	0.0256	0.0471	42	1641	99	**0.2979**	0.0250	0.0461
SC	80	1563	543	0.1284	0.0487	0.0706	87	1596	536	0.1396	0.0517	0.0755
WSD	82	1561	305	0.2119	0.0499	0.0808	91	1592	296	0.2351	0.0541	0.0879
NUCLE+ED	173	1470	411	**0.2962**	0.1053	**0.1554**	194	1489	390	**0.3322**	0.1153	**0.1712**
NUCLE+MORPH	163	1480	427	**0.2763**	0.0992	0.1460	182	1501	408	**0.3085**	0.1081	0.1601
NUCLE+POS	164	1479	365	**0.3100**	0.0998	0.1510	182	1501	347	**0.3440**	0.1081	0.1646
NUCLE+SC	162	1481	488	0.2492	0.0986	0.1413	181	1502	469	0.2785	0.1075	0.1552
NUCLE+WSD	163	1480	413	**0.2830**	0.0992	0.1469	181	1502	395	**0.3142**	0.1075	0.1602

Table 4: Evaluation of our correction systems over the original and revised NUCLE test set using the M^2 Scorer. Columns C, M and U show the number of correct, missed and unnecessary corrections suggested by each system. Results in bold show improvements over the baseline.

correct sentences and the target side contains their corrected versions. Our setup is similar to the one described by Yuan and Felice (2013) in that we train a PoS-factored phrase-based model (Koehn, 2010) using Moses (Koehn et al., 2007), Giza++ (Och and Ney, 2003) for word alignment and the IRSTLM Toolkit (Federico et al., 2008) for language modelling. However, unlike them, we do not optimise decoding parameters but use default values instead.

We build 11 different systems in total: a baseline system using only the NUCLE training set, one system per artificial corpus and other additional systems using combinations of the NUCLE training data and our artificial corpora. Each of these systems uses a single translation model that tackles all error types at the same time.

5 Results

Each system was evaluated in terms of precision, recall and F_1 on the NUCLE test data using the M^2 Scorer (Dahlmeier and Ng, 2012), the official evaluation script for the CoNLL 2013 shared task. Table 4 shows results of evaluation on the original test set (containing only one gold standard correction per error) and a revised version (which allows for alternative corrections submitted by participating teams).

Results reveal our ED and POS corpora are able to improve precision for both test sets. It is surprising, however, that the least informed dataset (ED) is one of the best performers although this seems reasonable if we consider it is the only dataset that includes artificial instances for all error types (see Table 1). Hybrid datasets containing the NUCLE training set plus an artificial corpus also generally improve precision, except for NUCLE+SC. It could be argued that the reason for this improvement is corpus size, since our hybrid datasets are double the size of each individual set, but the small differences in precision between the ED and POS datasets and their corresponding hybrid versions seem to contradict that hypothesis. In fact, results would suggest artificial and naturally occurring errors are not interchangeable but rather complementary.

The observed improvement in precision, however, comes at the expense of recall, for which none of the systems is able to beat the baseline. This contradicts results by Rozovskaya and Roth (2010a), who show their error inflation method increases recall, although this could be due to differences in the training paradigm and data. Still, results are encouraging since precision is generally preferred over recall in error correction scenarios (Yuan and Felice, 2013).

We also evaluated performance by error type on the original (Table 5) and revised (Table 6) test data using an estimation approach similar to the one in CoNLL 2013. Results show that the performance of each dataset varies by error type, suggesting that certain types of information are better suited for specific error types. In particular, we find that on the original test set, ED achieves the highest precision for article and determiners, WSD maximises precision for prepositions and SC achieves the highest recall and F_1. When using hybrid sets, results improve overall, with the highest precision being as follows: NUCLE+POS (ArtOrDet), NUCLE+ED (Nn), NUCLE+WSD

	ArtOrDet			Nn			Prep			SVA/Vform			Other		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**C**	**M**	**U**
NUCLE (b)	0.2716	0.1551	0.1974	0.4625	0.0934	0.1555	0.1333	0.0386	0.0599	0.2604	0.1016	0.1462	0	0	34
ED	**0.2813**	0.0391	0.0687	**0.6579**	0.0631	0.1152	0.0233	0.0032	0.0056	0.0000	0.0000	—	0	0	5
MORPH	0.1862	0.1058	0.1349	—	0.0000	—	0.0000	0.0000	—	0.1429	0.0041	0.0079	0	0	7
POS	0.0000	0.0000	—	0.4405	0.0934	0.1542	0.0000	0.0000	—	0.1515	0.0203	0.0358	0	0	10
SC	0.1683	0.0739	0.1027	—	0.0000	—	0.0986	**0.0932**	**0.0959**	0.0000	0.0000	—	0	0	21
WSD	0.2219	0.1029	0.1406	0.0000	0.0000	—	**0.1905**	0.0257	0.0453	0.1875	0.0122	0.0229	0	0	8
NUCLE+ED	**0.3185**	0.1348	0.1894	**0.5465**	**0.1187**	**0.1950**	0.1304	0.0386	0.0596	**0.2658**	0.0854	0.1292	0	0	35
NUCLE+MORPH	**0.2857**	0.1507	**0.1973**	0.4590	0.0707	0.1225	**0.1719**	0.0354	0.0587	**0.2817**	0.0813	0.1262	0	0	30
NUCLE+POS	**0.3384**	0.1290	0.1868	**0.4659**	0.1035	**0.1694**	**0.1884**	**0.0418**	**0.0684**	**0.2625**	0.0854	0.1288	0	0	29
NUCLE+SC	**0.2890**	0.1290	0.1784	0.4500	0.0682	0.1184	**0.1492**	**0.0868**	**0.1098**	**0.2836**	0.0772	0.1214	0	0	34
NUCLE+WSD	**0.3003**	0.1449	0.1955	**0.4667**	0.0707	0.1228	**0.1948**	**0.0482**	**0.0773**	**0.2632**	0.0813	0.1242	0	0	30

Table 5: Error type analysis of our correction systems over the original NUCLE test set using the M^2 Scorer. Columns C, M and U show the number of correct, missed and unnecessary corrections outside the main categories suggested by each system. Results in bold show improvements over the baseline.

	ArtOrDet			Nn			Prep			SVA/Vform			Other		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**C**	**M**	**U**
NUCLE (b)	0.3519	0.2026	0.2572	0.6163	0.1302	0.2150	0.2069	0.0682	0.1026	0.4105	0.1718	0.2422	0	0	34
ED	**0.4063**	0.0579	0.1014	**0.7297**	0.0684	0.1250	0.0465	0.0077	0.0132	0.1818	0.0183	0.0332	0	0	5
MORPH	0.2270	0.1311	0.1662	—	0.0000	—	0.0000	0.0000	—	0.2857	0.0092	0.0179	0	0	7
POS	0.0000	0.0000	—	0.5465	0.1169	0.1926	0.0000	0.0000	—	**0.4242**	0.0631	0.1098	0	0	10
SC	0.2112	0.0944	0.1305	—	0.0000	—	0.1088	**0.1221**	**0.1151**	0.0000	0.0000	—	0	0	21
WSD	0.2781	0.1313	0.1784	0.0000	0.0000	—	**0.2143**	0.0347	0.0598	0.2000	0.0138	0.0259	0	0	8
NUCLE+ED	**0.4334**	0.1849	**0.2592**	**0.7000**	**0.1552**	**0.2540**	0.1685	0.0575	0.0857	**0.4744**	0.1630	**0.2426**	0	0	35
NUCLE+MORPH	**0.3791**	0.2006	**0.2624**	**0.6308**	0.1017	0.1752	**0.2295**	0.0536	0.0870	**0.4714**	0.1454	0.2222	0	0	30
NUCLE+POS	**0.4601**	0.1761	0.2547	0.6087	**0.1383**	**0.2254**	**0.2424**	0.0613	0.0979	**0.4430**	0.1549	0.2295	0	0	29
NUCLE+SC	**0.3961**	0.1773	0.2450	0.6154	0.0993	0.1709	0.1844	**0.1250**	**0.1490**	**0.4848**	0.1410	0.2184	0	0	34
NUCLE+WSD	**0.3994**	0.1933	**0.2605**	**0.6308**	0.1017	0.1752	**0.2432**	**0.0690**	**0.1075**	**0.4667**	0.1535	0.2310	0	0	30

Table 6: Error type analysis of our correction systems over the revised NUCLE test set using the M^2 Scorer. Columns C, M and U show the number of correct, missed and unnecessary corrections outside the main categories suggested by each system. Results in bold show improvements over the baseline.

(Prep) and NUCLE+SC (SVA/Vform). As expected, the use of alternative annotations in the revised test set improves results but it does not reveal any qualitative difference between datasets.

Finally, when compared to other systems in the CoNLL 2013 shared task in terms of F_1, our best systems would rank 9th on both test sets. This would suggest that using an off-the-shelf SMT system trained on a combination of real and artificial data can yield better results than other machine learning techniques (Yi et al., 2013; van den Bosch and Berck, 2013; Berend et al., 2013) or rule-based approaches (Kunchukuttan et al., 2013; Putra and Szabo, 2013; Flickinger and Yu, 2013; Sidorov et al., 2013).

6 Conclusions

This paper presents early results on the generation and use of artificial errors for grammatical error correction. Our approach uses conditional probabilities derived from an ESL error-annotated corpus to replicate errors in native error-free data. Unlike previous work, we propose using linguistic information such as PoS or sense disambiguation to refine the contexts where errors occur and thus replicate them more accurately. We use five different types of information to generate our artificial corpora, which are later evaluated in isolation as well as coupled to the original ESL training data.

General results show error distributions and PoS information produce the best results, although this varies when we analyse each error type separately. These results should allow us to generate errors more efficiently in the future by using the best approach for each error type.

We have also observed that precision improves at the expense of recall and this is more pronounced when using purely artificial sets. Finally, artificially generated errors seem to be a complement rather than an alternative to genuine data.

7 Future work

There are a number of issues we plan to address in future research, as described below.

Scaling up artificial data

The experiments presented here use a small and manually selected collection of Wikipedia articles.

However, we plan to study the performance of our systems as corpus size is increased. We are currently using a larger selection of Wikipedia articles to produce new artificial datasets ranging from 50K to 5M sentences. The resulting corpora will be used to train new error correction systems and study how precision and recall vary as more data is added during the training process, similar to Cahill et al. (2013).

Reducing differences between datasets

As shown in Table 1, we are unable to produce the same set of errors for each different type of information. This is a limitation of our conditional probabilities which encode different information in each case. In consequence, comparing overall results between datasets seems unfair as they do not target the same error types. In order to overcome this problem, we will define new probabilities so that we can generate the same types of error in all cases.

Exploring larger contexts

Our current probabilities model error contexts in a limited way, mostly by considering relations between two or three words (e.g. article+noun, verb+preposition+noun, etc.). In order to improve error injection, we will define new probabilities using larger contexts, such as P(source=verb|target=verb, subject_class, auxiliary_verbs, object_class) for verb form errors. Using more specific contexts can also be useful for correcting complex error types, such as the use of pronouns, which often requires analysing coreference chains.

Using new linguistic information

In this work we have used five types of linguistic information. However, we believe other types of information and their associated probabilities could also be useful, especially if we aim to correct more error types. Examples include spelling, grammatical relations (dependencies) and word order (syntax). Additionally, we believe the use of semantic role labels can be explored as an alternative to semantic classes, as they have proved useful for error correction (Liu et al., 2010).

Mixed error generation

In our current experiments, each artificial corpus is generated using only one type of information at a time. However, having found that certain types of information are more suitable than others for correcting specific error types (see Tables 5 and 6), we believe better artificial corpora could be created by generating instances of each error type using only the most appropriate linguistic information.

Acknowledgments

We would like to thank Prof Ted Briscoe for his valuable comments and suggestions as well as Cambridge English Language Assessment, a division of Cambridge Assessment, for supporting this research.

References

Gabor Berend, Veronika Vincze, Sina Zarrieß, and Richárd Farkas. 2013. Lfg-based features for noun number and article grammatical errors. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 62–67, Sofia, Bulgaria, August. Association for Computational Linguistics.

Steven Bird, Edward Loper, and Ewan Klein. 2009. *Natural Language Processing with Python*. O'Reilly Media Inc.

Ted Briscoe, John Carroll, and Rebecca Watson. 2006. The second release of the RASP system. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, COLING-ACL '06, pages 77–80, Sydney, Australia. Association for Computational Linguistics.

Chris Brockett, William B. Dolan, and Michael Gamon. 2006. Correcting ESL Errors Using Phrasal SMT Techniques. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 249–256, Sydney, Australia, July. Association for Computational Linguistics.

Aoife Cahill, Nitin Madnani, Joel Tetreault, and Diane Napolitano. 2013. Robust systems for preposition error correction using wikipedia revisions. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 507–517, Atlanta, Georgia, June. Association for Computational Linguistics.

Daniel Dahlmeier and Hwee Tou Ng. 2012. Better evaluation for grammatical error correction. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL 2012, pages 568 – 572, Montreal, Canada.

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 2013. Building a Large Annotated Corpus of Learner English: The NUS Corpus of Learner English. In *Proceedings of the 8th Workshop on Innovative Use of NLP for Building Educational Applications*, BEA 2013, pages 22–31, Atlanta, Georgia, USA, June. Association for Computational Linguistics.

Nava Ehsan and Heshaam Faili. 2013. Grammatical and context-sensitive error correction using a statistical machine translation framework. *Software: Practice and Experience*, 43(2):187–206.

Marcello Federico, Nicola Bertoldi, and Mauro Cettolo. 2008. IRSTLM: an open source toolkit for handling large scale language models. In *Proceedings of the 9th Annual Conference of the International Speech Communication Association*, INTERSPEECH 2008, pages 1618–1621, Brisbane, Australia, September. ISCA.

Dan Flickinger and Jiye Yu. 2013. Toward more precision in correction of grammatical errors. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 68–73, Sofia, Bulgaria, August. Association for Computational Linguistics.

Jennifer Foster and Øistein Andersen. 2009. Generrate: Generating errors for use in grammatical error detection. In *Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 82–90, Boulder, Colorado, June. Association for Computational Linguistics.

William A. Gale, Kenneth W. Church, and David Yarowsky. 1992. One sense per discourse. In *Proceedings of the workshop on Speech and Natural Language*, HLT '91, pages 233–237, Harriman, New York. Association for Computational Linguistics.

Kenji Imamura, Kuniko Saito, Kugatsu Sadamitsu, and Hitoshi Nishikawa. 2012. Grammar error correction using pseudo-error sentences and domain adaptation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 388–392, Jeju Island, Korea, July. Association for Computational Linguistics.

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 on Association for Computational Linguistics - Volume 2*, ACL '03, pages 145–148, Sapporo, Japan. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Philipp Koehn. 2010. *Statistical Machine Translation*. Cambridge University Press, New York, NY, USA, 1st edition.

Anoop Kunchukuttan, Ritesh Shah, and Pushpak Bhattacharyya. 2013. Iitb system for conll 2013 shared task: A hybrid approach to grammatical error correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 82–87, Sofia, Bulgaria, August. Association for Computational Linguistics.

Claudia Leacock, Martin Chodorow, Michael Gamon, and Joel Tetreault. 2010. *Automated Grammatical Error Detection for Language Learners*. Morgan and Claypool Publishers.

John Lee and Stephanie Seneff. 2008. An analysis of grammatical errors in non-native speech in English. In Amitava Das and Srinivas Bangalore, editors, *Proceedings of the 2008 IEEE Spoken Language Technology Workshop*, SLT 2008, pages 89–92, Goa, India, December. IEEE.

Xiaohua Liu, Bo Han, Kuan Li, Stephan Hyeonjun Stiller, and Ming Zhou. 2010. SRL-based verb selection for ESL. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1068–1076, Cambridge, MA, October. Association for Computational Linguistics.

George A. Miller. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, November.

Tomoya Mizumoto, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. 2011. Mining Revision Log of Language Learning SNS for Automated Japanese Error Correction of Second Language Learners. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 147–155, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

Hwee Tou Ng, Siew Mei Wu, Yuanbin Wu, Christian Hadiwinoto, and Joel Tetreault. 2013. 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, August. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Comput. Linguist.*, 29(1):19–51, March.

Ted Pedersen and Varada Kolhatkar. 2009. WordNet::SenseRelate::AllWords: a broad coverage word sense tagger that maximizes semantic relatedness. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Demonstration Session*, NAACL-Demonstrations '09, pages 17–20, Boulder, Colorado. Association for Computational Linguistics.

Desmond Darma Putra and Lili Szabo. 2013. Uds at conll 2013 shared task. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 88–95, Sofia, Bulgaria, August. Association for Computational Linguistics.

Alla Rozovskaya and Dan Roth. 2010a. Training paradigms for correcting errors in grammar and usage. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 154–162, Los Angeles, California. Association for Computational Linguistics.

Alla Rozovskaya and Dan Roth. 2010b. Generating confusion sets for context-sensitive error correction. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 961–970, Cambridge, Massachusetts. Association for Computational Linguistics.

Alla Rozovskaya and Dan Roth. 2011. Algorithm selection and model adaptation for ESL correction tasks. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 924–933, Portland, Oregon. Association for Computational Linguistics.

Alla Rozovskaya, Mark Sammons, and Dan Roth. 2012. The UI system in the HOO 2012 shared task on error correction. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 272–280, Montreal, Canada. Association for Computational Linguistics.

Alla Rozovskaya, Kai-Wei Chang, Mark Sammons, and Dan Roth. 2013. 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, August. Association for Computational Linguistics.

Grigori Sidorov, Anubhav Gupta, Martin Tozer, Dolors Catala, Angels Catena, and Sandrine Fuentes. 2013. Rule-based system for automatic grammar correction using syntactic n-grams for english language learning (l2). In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 96–101, Sofia, Bulgaria, August. Association for Computational Linguistics.

Jonas Sjöbergh and Ola Knutsson. 2005. Faking errors to avoid making errors: Very weakly supervised learning for error detection in writing. In *Proceedings of RANLP 2005*, pages 506–512, Borovets, Bulgaria, September.

Toshikazu Tajiri, Mamoru Komachi, and Yuji Matsumoto. 2012. Tense and aspect error correction for esl learners using global context. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 198–202, Jeju Island, Korea, July. Association for Computational Linguistics.

Joel Tetreault, Daniel Blanchard, and Aoife Cahill. 2013. A Report on the First Native Language Identification Shared Task. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 48–57, Atlanta, Georgia, June. Association for Computational Linguistics.

Antal van den Bosch and Peter Berck. 2013. Memory-based grammatical error correction. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 102–108, Sofia, Bulgaria, August. Association for Computational Linguistics.

Bong-Jun Yi, Ho-Chang Lee, and Hae-Chang Rim. 2013. Kunlp grammatical error correction system for conll-2013 shared task. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 123–127, Sofia, Bulgaria, August. Association for Computational Linguistics.

Zheng Yuan and Mariano Felice. 2013. Constrained grammatical error correction using statistical machine translation. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task*, pages 52–61, Sofia, Bulgaria, August. Association for Computational Linguistics.