Statistical Machine Translation for Grammatical Error Correction

Arvind Srinivasan
Columbia University
vs2371@columbia.edu

Louis Cialdella
Columbia University
lmc2179@columbia.edu

Abstract

This document contains the instructions for preparing a camera-ready manuscript for the proceedings of NAACL HLT 2012. The document itself conforms to its own specifications, and is therefore an example of what your manuscript should look like. Authors are asked to conform to all the directions reported in this document. Also included are instructions for submission to the conference, which are similar to the camera-ready instructions but remove author-identifying information.

1 Background

This paper details our experiments in using Statistical Machine translation methods for error detection in English as part of the Conll-13 challenge. The challenge involved correction of errors made by Singaporean students of English

2 Task + Corpus

This project was initiated as a submission to the ConLL 2013 Shared Task: Grammatical Error Correction [1]. As a result, all training and test data was from a provided corpus, the National University Singapore Corpus of Learner English, or NUCLE, and trained systems were scored with the provided NUS MaxMatch, or M^2 scorer.

The corpus itself is comprised of 1400 annotated essays written by english-second-language Singaporean students. The annotations provide an error type and correction, from the below types of errors:

tag	category
Vt	Verb tense
Vm	Verb modal
V0	Missing verb
Vform	Verb form
SVA	Subject-verb-agreement
ArtOrDet	Article or Determiner
Nn	Noun number
Npos	Noun possesive
Pform	Pronoun form
Pref	Pronoun reference
Prep	Preposition
Wci	Wrong collocation/idiom
Wa	Acronyms
Wform	Word form
Wtone	Tone
Srun	Runons, comma splice
Smod	Dangling modifier
Spar	Parallelism
Sfrag	Fragment
Ssub	Subordinate clause
WOinc	Incorrect sentence form
WOadv	Adverb/adjective position
Trans	Link word/phrases
Mec	Punctuation, capitalization, spelling, typos
Rloc	Local redundancy
Cit	Citation
Others	Other errors
Um	Unclear meaning (cannot be corrected)

For the initial approach to this task, however, we looked to train a system that would identify a set of five errors: SVA, ArtOrDet, Nn, VForm, and Prep. The total corpus amounted to 67372 sentences, of which 15,000 were noisy (references, urls, etc).

[1] http://www.comp.nus.edu.sg/ nlp/conll13st.html

Of that, 11288 sentences had annotations with those tags, leaving a fairly small dataset on which to train. After the time of this writing, ConLL has released an additional dataset - the blind test data with the gold references, so we anticipate that this dataset will grow significantly.

3 Problem Background: Types of error

We begin by motivating the approach we take by considering the problem in a little more depth. Two of the most common types of error are article errors and preposition errors. Article errors are most common for speakers of languages which themselves lack any sort of article (some very common examples are Chinese, Japanese, and Russian), though they are naturally present in the writings of most all students. Similarly, preposition errors account for a large number of errors, due to the vast complexity of the English preposition system. Preposition use is highly particular, and is often governed by the context in which the preposition appears (making it a good candidate for phrase based correction).

An additional major source of error is that of collocation (or idiomatic) errors. These arise when there is a strong association between words even though other choices might be semantically or syntactically correct (for example, strong computer makes sense, but powerful computer is idiomatically rendered). Additionally, this type of error encompasses stock idiomatic phrases where the individual words are not obviously related to the actual meaning (to hit the nail on the head, to kick the bucket, etc). These phrases are usually non-modifiable in many contexts or cannot be modified and retain their meaning unless great care is used. While rule-based approaches lead to extremely complex definitions, a statistical approach is well suited to collocation errors since it provides an easy and concrete way of taking into account the relationships between words and/or whole idiomatic phrases.

4 Baseline

Our baseline system used Moses to ”translate” between a test corpus of original essays and the counterpart essays with all the annotations applied. This ”flattened” corpus was split into 58000 training sentences, 2000 sentences for tuning, and 8000 test sentences.

Moses used Giza++ to align the sentences, with default heuristics and reordering models (grow-diag-final, msd-bidirectional-fe). The tuning used MERT with default features.

To score the baseline, we use BLEU as well as the M^2 scorer. The M^2 scorer scores precision, accuracy, and F1 against the annotations, rather than the text itself. Though the conference version of M^2 is case-sensitive, we use a case-insensitive version for simplicity - recasing in English is a trivial task. Though both of these scorers support multi-reference evaluation, the corpus itself only has a single-reference gold reference file. Clearly, there are multiple equally fluent machine-generated correction candidates, even within the phrase table generated by the small training corpus, so we think this is an area that needs to be explored in terms of test corpus augmentation.

In reality, the optimality of a ”correction” would be gauged by fluency and grammatical cohesion of the final generation, so neither Bleu nor M^2 adequately capture the ideal result - an ideal result would likely be modeled by a combination of the two, with multiple references.

5 Issues with Baseline

The baseline displayed atrocious performance across the board. Two major and easily observable issues arise: the number of useless phrases in the phrase table, and the bias of phrases towards not making changes.

Firstly, we note that the vast majority of phrases in the phrase table are entirely unhelpful in decoding. For example, the number of possible translations of ”(” is obscenely large, having hundreds of entries. Of these, only a handful are useful, and even then the most common substitution should still be ”(”, as no errors using the left paren character occur anyway. This happens for virtually every piece of punctuation and trivial phrase, and does nothing except slow down decoding and cause spurious translation issues during decoding. We deal with a solution to this in the phrase table pruning section.

A second issue is that the phrase table entries are biased towards not making changes when decoding happens. That is, for a given phrase, the phrase table causes there to be a much larger chance of said phrase remaining unchanged in the final product, rather than changing and potentially correcting an error. A fairly typical PT entry indicative of this issue is:

, according to ——— , according to the ——— 0.166667 0.99065 0.0384615 0.578006 2.718 ——— 0-0 1-1 2-2 ——— 6 26 1

, according to ——— , according to ——— 0.961538 0.99065 0.961538 0.986281 2.718 ——— 0-0 1-1 2-2 ——— 26 26 25

We note here that the bottom phrase (which makes no changes) has a probability of 0.961538, while the top phrase has a probability of 0.0384615. We talk about our approach to this in the section on downsampling.

Additionally, we run into the problem of the BLEU metric, which causes problems with both tuning and evaluation. In this task, the BLEU metric used by Moses is quite a poor indicator of success. BLEU only checks for particular N-Grams in the reference, and in this task there is only a single reference sentence. This means that BLEU measures only N-grammatic distance to the supplied reference, which makes no guarantee of fluency. That is, a given change might drastically improve a sentence and make an otherwise incorrect sentence mostly or completely correct, but this may not be reflected in the BLEU score if it does not completely line up with the reference sentence. This leads to both strange tuning artifacts (such as the over-tuning described above) and difficult-to-interpret evaluation results.

6 Corpus cleaning and Datasets

The first immediate issue with the baseline phrase table was that references and urls generated a large amount of noise in the correction data. Since we used the wordpunct NLTK tokenization scheme, urls in particular were inconsistently treated as multiple words, creating noisy alignments as well. Thus, the first cleaning step was to eliminate references completely, re-adding them after the correction. We accomplished this with a simple reegex substitution.

Additionally, to experiment on the domain sensitivity of our system, we split the corpus by essay topic. This was particularly necessary because one of the challenges of the shared task was to submit system output for topics both inside and outside the training data. However, since the topics were not available a priori, we used an online version of the Latent Dirichlet Allocation (LDA) algorithm to generate a topic model, and then a K-Means clustering implementation to split the documents by those topics. We were then able to generate a ”held out” dataset with 21572 training sentences (no references) with all but one topic, and 8695 test sentences (no references) with the remaining topic. This experiment showed that the NUCLE essay topics were largely similar in content, and that domain sensitivity is still a concern that ought to be tested by a held-out test set with a topic further removed from the training corpus.

To experiment with the impact of zero-annotation sentences in the training corpus, we also generated datasets without correct sentences. This left a dataset of 11288 sentences, of which we used 10000 to train and 288 to tune.

To run stemming experiments, we used the stemmers bundled with NLTK. We stemmed and lemmatized the two corpora mentioned above with the Lancaster and Snowball stemmers and the WordNet lemmatizer to experiment with various degrees of stemming aggressiveness.

We used a Moses (SRILM) ngram language model trained on the test and dev datasets. Given the topic closeness between the test and train sets and the relatively small size of the test set, there was not a significant OOV problem, and experiments with big language models trained on the Brown Corpus did not yield any benefits to either of the scoring metrics.

7 Approach

7.1 Significance Testing

7.2 Downsampling

7.3 Stemming

7.4 NoCorrect - Only Errors

8 Results

All the experiments we ran achieved consistently high Bleu scores, but this includes the sentences that were already correct and mapped to another correct sentence. When the test set was scored with BLEU without any modifications, it achieved a BLEU score of of 87.32, 95.8/90.6/85.5/80.6.

Experiment	**Bleu**
Baseline	*96.34, 98.7/97.1/95.6/94.1*
Lancaster	*73.04, 88.2/77.6/68.6/60.6*
WordNet	*86.48, 94.1/88.8/84.1/79.6*
HeldOut	*96.81, 98.9/97.5/96.2/94.9*
NoCorrect	*96.46, 98.7/97.2/95.7/94.3*
Stem+NoCorr	*88.26, 95.8/90.8/86.0/81.6*
HeldOut+Sig	*96.80, 98.9/97.5/96.1/94.8*
NoCorr+Sig	*95.59, 98.5/96.5/94.6/92.8*
Lanc+Sig	*89.01, 96.3/91.2/87.0/82.9*
Word+Sig	*92.87, 97.4/94.4/91.6/88.8*

With the M^2 scorer, the different experiments yielded marginal improvements. In this case, the starting point for the score was 0.00, since the precision/recall/f1 were evaluated against only the generated and gold annotations.

M^2	*Precision*	Recall	F1
Baseline	*0*	0	0
Lancaster	*0.0215*	0.1915	0.0387
WordNet	*0.049*	0.1773	0.0768
HeldOut	*0.0116*	0.0663	0.0198
NoCorrect	*0.0838*	0.2555	0.1262
Stem+NoCorr	*0.0624*	0.3086	0.1038
HeldOut+Sig	*0.0086*	0.0312	0.0134
NoCorr+Sig	*0.0268*	0.1031	0.0425
Lanc+Sig	*0.0073*	0.0312	0.0118
Word+Sig	*0.0221*	0.0906	0.0355

For the significance testing experiments, a significant amount of the phrase table was pruned of statistically insignificant entries.

M2	*% of PT pruned*
HeldOut+Sig	*56.90%*
NoCorr+Sig	*53.17%*
Lanc+Sig	*51.81%*
Word+Sig	*57.57%*

9 Observations

10 Future work

10.1 Footnotes

Footnotes: Put footnotes at the bottom of the page. They may be numbered or referred to by asterisks or other symbols.[2] Footnotes should be separated from the text by a line.[3] Footnotes should be in 9 point font.

10.2 Graphics

Illustrations: Place figures, tables, and photographs in the paper near where they are first discussed, rather than at the end, if possible. Wide illustrations may run across both columns and should be placed at the top of a page. Color illustrations are discouraged, unless you have verified that they will be understandable when printed in black ink.

Captions: Provide a caption for every illustration; number each one sequentially in the form: “Figure 1. Caption of the Figure.” “Table 1. Caption of the Table.” Type the captions of the figures and tables below the body, using 10 point text.

11 Length of Submission

The NAACL HLT 2012 main conference accepts submissions of long papers and short papers. The maximum length of a long paper manuscript is eight (8) pages of content and two (2) additional pages of references *only* (appendices count against the eight pages, not the additional two pages). The maximum length of a short paper manuscript is four (4) pages including references. For both long and short papers, all illustrations, references, and appendices must be accommodated within these page limits,

[2] This is how a footnote should appear.

[3] Note the line separating the footnotes from the text.

observing the formatting instructions given in the present document. Papers that do not conform to the specified length and formatting requirements are subject to be rejected without review.

12 Double-blind review process

As the reviewing will be blind, the paper must not include the authors' names and affiliations. Furthermore, self-references that reveal the author's identity, e.g., ”We previously showed (Smith, 1991) ...” must be avoided. Instead, use citations such as ”Smith previously showed (Smith, 1991) ...” Papers that do not conform to these requirements will be rejected without review. In addition, please do not post your submissions on the web until after the review process is complete.

Acknowledgments

Do not number the acknowledgment section.

References

Alfred V. Aho and Jeffrey D. Ullman. 1972. *The Theory of Parsing, Translation and Compiling*, volume 1. Prentice-Hall, Englewood Cliffs, NJ.

American Psychological Association. 1983. *Publications Manual*. American Psychological Association, Washington, DC.

Association for Computing Machinery. 1983. *Computing Reviews*, 24(11):503–512.

Ashok K. Chandra, Dexter C. Kozen, and Larry J. Stockmeyer. 1981. Alternation. *Journal of the Association for Computing Machinery*, 28(1):114–133.

Dan Gusfield. 1997. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, Cambridge, UK.