Exploring Grammatical Error Correction with Not-So-Crummy Machine Translation*

Nitin Madnani **Joel Tetreault**
Educational Testing Service
Princeton, NJ, USA
{nmadnani,jtetreault}@ets.org

Martin Chodorow
Hunter College of CUNY
New York, NY, USA
martin.chodorow@hunter.cuny.edu

Abstract

To date, most work in grammatical error correction has focused on targeting specific error types. We present a probe study into whether we can use round-trip translations obtained from Google Translate via 8 different pivot languages for **whole-sentence** grammatical error correction. We develop a novel alignment algorithm for combining multiple round-trip translations into a lattice using the TERp machine translation metric. We further implement six different methods for extracting whole-sentence corrections from the lattice. Our preliminary experiments yield fairly satisfactory results but leave significant room for improvement. Most importantly, though, they make it clear the methods we propose have strong potential and require further study.

1 Introduction

Given the large and growing number of non-native English speakers around the world, detecting and correcting grammatical errors in learner text currently ranks as one of the most popular educational NLP applications. Previously published work has explored the effectiveness of using **round-trip machine translation** (translating an English sentence to some foreign language F, called the *pivot*, and then translating the F language sentence back to English) for correcting preposition errors (Hermet and Désilets, 2009). In this paper, we present a pilot study that explores the effectiveness of extending this approach to whole-sentence grammatical error correction.

Specifically, we explore whether using the concept of round-trip machine translation via *multiple*—rather than single—pivot languages has the potential of correcting most, if not all, grammatical errors present in a sentence. To do so, we develop a round-trip translation framework using the Google Translate API. Furthermore, we propose a novel combination algorithm that can combine the evidence present in multiple round-trip translations and increase the likelihood of producing a whole-sentence correction. Details of our methodology are presented in §3 and of the dataset we use in §4. Since this work is of an exploratory nature, we conduct a detailed error analysis and present the results in §5. Finally, §6 summarizes the contributions of this pilot study and provides a discussion of possible future work.

2 Related Work

To date, most work in grammatical error detection has focused on targeting specific error types (usually prepositions or article errors) by using rule-based methods or statistical machine-learning classification algorithms, or a combination of the two. Leacock et al. (2010) present a survey of the common approaches. However, targeted errors such as preposition and determiner errors are just two of the many types of grammatical errors present in non-native writing. One of the anonymous reviewers for this paper makes the point eloquently: "*Given the frequent complexity of learner errors, less holistic, error-type specific approaches are often unable to*

*cf. *Good Applications for Crummy Machine Translation*. Ken Church & Ed Hovy. Machine Translation, 8(4). 1993

The 7th Workshop on the Innovative Use of NLP for Building Educational Applications, pages 44–53,
Montréal, Canada, June 3-8, 2012.

disentangle compounded errors of style and grammar." Below we discuss related work that uses machine translation to address targeted errors and some recent work that also focused on whole-sentence error correction.

Brockett et al. (2006) use information about mass noun errors from a Chinese learner corpus to engineer a "parallel" corpus with sentences containing mass noun errors on one side and their corrected counterparts on the other. With this parallel corpus, the authors use standard statistical machine translation (SMT) framework to learn a translation (correction) model which can then be applied to unseen sentences containing mass noun errors. This approach was able to correct almost 62% of the errors found in a test set of 150 errors. In our approach, we do not treat correction directly as a translation problem but instead rely on an MT system to round-trip translate an English sentence back to English.

Park and Levy (2011) use a noisy channel model to achieve whole-sentence grammar correction; they learn a noise model from a dataset of errorful sentences but do not rely on SMT. They show that the corrections produced by their model generally have higher n-gram overlap with human-authored reference corrections than the original errorful sentences.

The previous work that is most directly relevant to our approach is that of Hermet and Désilets (2009) who focused only on sentences containing pre-marked preposition errors and generated a *single* round-trip translation for such sentences via a single pivot language (French). They then simply posited this round-trip translation as the "correction" for the original sentence. In their evaluation on sentences containing 133 unique preposition errors, their round-trip translation system was able to correct 66.4% of the cases. However, this was outperformed by a simple method based on web counts (68.7%). They also found that combining the round-trip method with the web counts method into a hybrid system yielded higher performance (82.1%).

In contrast, we use multiple pivot languages to generate several round-trip translations. In addition, we use a novel alignment algorithm that allows us to combine different parts of different round-trip translations and explore a whole new set of corrections that go beyond the translations themselves. Finally, we do not restrict our analysis to any single type of error. In fact, our test sentences contain several different types of grammatical errors.

Outside of the literature on grammatical error detection, our combination approach is directly related to the research on machine translation system combination wherein translation hypotheses produced by different SMT systems are combined to allow the extraction of a better, combined hypothesis (Bangalore et al., 2001; Rosti et al., 2007; Feng et al., 2009). However, our combination approach is different in that all the round-trip translations are produced by a single system but via different pivot languages.

Finally, the idea of combining multiple surface renderings with the same meaning has also been explored in paraphrase generation. Pang et al. (2003) propose an algorithm to align sets of parallel sentences driven entirely by the syntactic representations of the sentences. The alignment algorithm outputs a merged lattice from which lexical, phrasal, and sentential paraphrases could simply be read off. Barzilay and Lee (2003) cluster topically related sentences into slotted word lattices by using multiple sequence alignment for the purpose of downstream paraphrase generation from comparable corpora. More recently, Zhao et al. (2010) perform round-trip translation of English sentences via different pivot languages and different off-the-shelf SMT systems to generate candidate paraphrases. However, they do not combine the candidate paraphrases in any way. A detailed survey of paraphrase generation techniques can be found in (Androutsopoulos and Malakasiotis, 2010) and (Madnani and Dorr, 2010).

3 Methodology

The basic idea underlying our error correction technique is quite simple: if we can automatically generate alternative surface renderings of the meaning expressed in the original sentence and then pick the one that is most fluent, we are likely to have picked a version of the sentence in which the original grammatical errors have been fixed.

In this paper, we propose generating such alternative formulations using statistical machine translation. For example, we take the original sentence E and translate it to Chinese using the Google Trans-

Original	Both experience and books are very important *about living*.
Swedish	Both experience and books are very important in live.
Italian	Both books are very important experience and life.
Russian	**And the experience, and a very important book about life.**
French	Both experience and the books are very important in life.
German	Both experience and books are very important about life.
Chinese	Related to the life experiences and the books are very important.
Spanish	Both experience and the books are very important about life.
Arabic	Both experience and books are very important for life.

Figure 1: Illustrating the deficiency in using an n-gram language model to select one of the 8 round-trip translations as the correction for the Original sentence. The grammatical errors in the Original sentence are shown in italics. The round-trip translation via Russian is chosen by a 5-gram language model trained on the English gigaword corpus even though it changes the meaning of the original sentence entirely.

late API. We then take the resulting Chinese sentence C and translate it back to English. Since the translation process is designed to be meaning-preserving, the resulting **round-trip translation** E' can be seen as an alternative formulation of the original sentence E. Furthermore, if additional pivot languages besides Chinese are used, several alternative formulations of E can be generated. We use 8 different pivot languages: Arabic, Chinese, Spanish, French, Italian, German, Swedish, Russian. We chose these eight languages since they are frequently used in SMT research and shared translation tasks. To obtain the eight round-trip translations via each of these pivot languages, we use the Google Translate research API.[1]

3.1 Round-Trip Translation Combination

Once the translations are generated, an obvious solution is to pick the most fluent alternative, e.g., using an n-gram language model. However, since the language model has no incentive to preserve the meaning of the sentence, it is possible that it might pick a translation that changes the meaning of the original sentence entirely. For example, consider the sentence and its round-trip translations shown in Figure 1. For this sentence, a 5-gram language model trained on gigaword picks the Russian round-trip translation simply because it has n-grams that were seen more frequently in the English gigaword corpus.

Given the deficiencies in statistical phrase-based translation, it is also possible that no single round-trip translation fixes all of the errors. Again, consider Figure 1. None of the 8 round-trip translations is error-free itself. Therefore, the task is more complex than simply selecting the right round-trip translation. We posit that a better approach will be to combine the evidence of correction produced by each independent translation model and increase the likelihood of producing a final whole-sentence correction. Additionally, by engineering such a combination, we increase the likelihood that the final correction will preserve the meaning of the original sentence.

In order to combine the round-trip translations, we developed a heuristic alignment algorithm that uses the TERp machine translation metric (Snover et al., 2009). The TERp metric takes a pair of sentences and computes the least number of edit operations that can be employed to turn one sentence into the other.[2] As a by-product of computing the edit sequence, TERp produces an *alignment* between the two sentences where each alignment link is defined by an edit operation. Figure 2 shows an example of the alignment produced by TERp between the original sentence from Figure 1 and its Russian round-trip translation. Note that TERp also allows shifting words and phrases in the second sentence in order to obtain a smaller edit cost (as indicated by the asterisk next to the word *book* which has shifted from its original position in the Russian round-trip translation).

Our algorithm starts by treating the original sentence as the backbone of a lattice. First, it cre-

[1] `http://research.google.com/university/translate/`

[2] Edit operations in TERp include matches, substitutions, insertion, deletions, paraphrase, synonymy and stemming.

ates a node for each word in the original sentence and creates edges between them with a weight of 1. Then, for each of the round-trip translations, it computes its TERp alignment with the original sentence and aligns it to the backbone based on the edit operations in the alignment. Specifically, each insertion, substitution, stemming, synonymy and paraphrase operation lead to creation of new nodes that essentially provide an alternative formulation for the aligned substring from the backbone. Any duplicate nodes are merged. Finally, edges produced by different translations between the same pairs of nodes are merged and their weights added. Figure 3(a) shows how our algorithm aligns the Russian round-trip translation from Figure 1 to the original sentence using the TERp alignment from Figure 2. Figure 3(b) shows the final lattice produced by our algorithm for the sentence and all the round-trip translations from Figure 1.

```
           -- and         [I]
      both -- the         [S]
experience -- experience  [M]
           -- ,           [I]
       and -- and         [M]
     books -- book        [T] [*]
       are -- a           [S]
      very -- very        [M]
 important -- important   [M]
     about -- about       [M]
    living -- life        [Y]
         . -- .           [M]
```

Figure 2: The alignment produced by TERp between the original sentence from Figure 1 and its Russian round-trip translation. The alignment operations are indicated in square brackets after each alignment link: **I**=insertion, **M**=match, **S**=substitution, **T**=stemming and **Y**=WordNet synonymy. The asterisk next to the work *book* denotes that TERp chose to shift its position before computing an edit operation for it.

3.2 Correction Generation

For each original sentence, we computed six possible corrections from the round-trip translations and the combined lattice:

1. **Baseline LM (B)**. The most fluent round-trip translation out of the eight as measured by a 5-gram language model trained on the English gigaword corpus.

2. **Greedy (G)**. A path is extracted from the TERp lattice using a greedy best-first strategy at each node, i.e., at each node, the outgoing edge with the largest weight is followed.

3. **1-Best (1)**: The shortest path is extracted from the TERp lattice by using the OpenFST toolkit.[3]. This method assumes that, like **G**, the combined evidence from the round-trip translations itself is enough to produce a good final correction and no external method for measuring fluency is required.[4]

4. **LM Re-ranked (L)**. An n-best (n=20) list is extracted from the lattice using the OpenFST toolkit and re-ranked using the 5-gram gigaword language model. The 1-best reranked item is then extracted as the correction. This method assumes that an external method of measuring fluency—the 5-gram language model—can help to bring the most grammatical correction to the top of the n-best list.

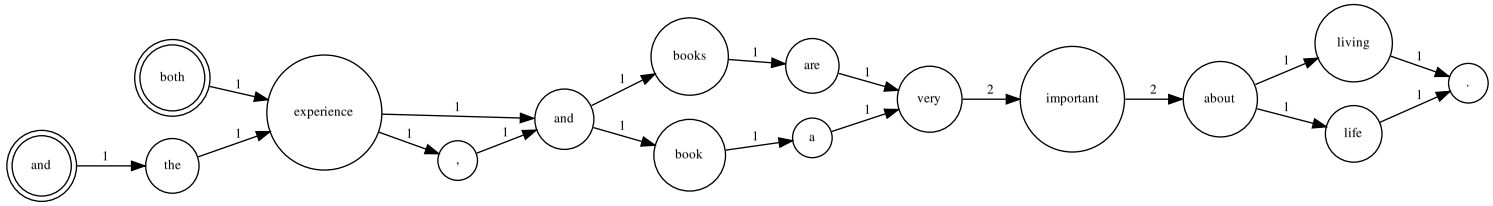
5. **Product Re-ranked (P)**. Same as L except the re-ranking is done based on the product of the cost of each hypothesis in the n-best list and the language model score, i.e., both the evidence from the round-trip translations and the language model is weighted equally.

6. **Full LM Composition (C)**. The edge weights in the TERp lattice are converted to probabilities. The lattice is then composed with a trigram finite state language model (trained on a corpus of $100,000$ high-scoring student essays).[5] The shortest path through the composed lattice is then extracted as the correction. This method assumes that using an n-gram language model during the actual search process is better than using it as a post-processing tool on an already extracted n-best list, such as for **L** and **P**.

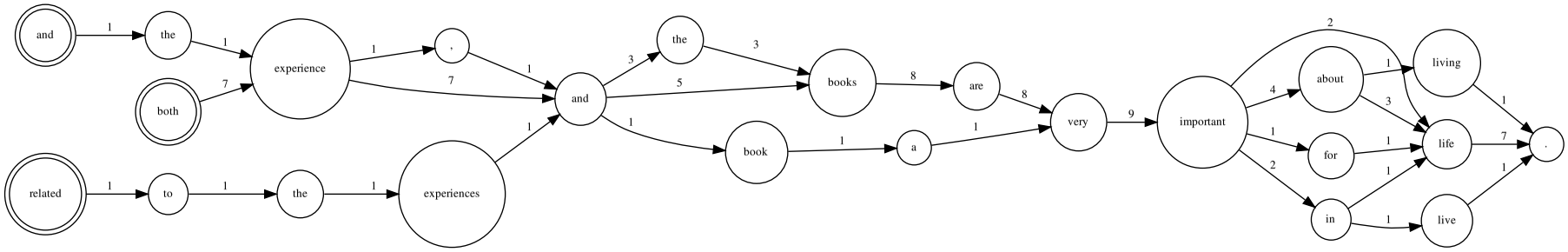
[3] `http://www.openfst.org/`

[4] Note that the edge weights in the lattice must be converted into costs for this method (we do so by multiplying the weights by -1).

[5] We adapted the code available at `http://www.ling.ohio-state.edu/~bromberg/ngramcount/ngramcount.html` to perform the LM composition.



(a)



(b)

Original (O)	Both experience and books are very important *about living*.
Baseline LM (B)	And the experience, and a very important book about life.
Greedy (G)	Both experience and books are very important about life.
1-best (1)	Both experience and the books are very important about life.
LM Re-ranked (L)	And the experience and the books are very important in life.
Product Re-ranked (P)	Both experience and books are very important about life.
LM Composition (C)	**Both experience and books are very important in life.**

(c)

Figure 3: (a) shows the output of our alignment algorithm for the Russian round-trip translation from Figure 1. (b) shows the final TERp lattice after aligning all eight round-trip translations from Figure 1. (c) shows the corrections for the original sentence (O) produced by the six techniques discussed in 3.2. The correction produced by the Full LM Composition technique (C) fixes both the errors in the original sentence.

No. of Errors	Sentences	Avg. Length
1	61	14.4
2	45	19.9
3	29	24.2
4	14	29.4
> 4	13	38.0

Table 1: The distribution of grammatical errors for the 162 errorful sentences.

Figure 3(c) shows these six corrections as computed for the sentence from Figure 1.

4 Corpus

To assess our system, we manually selected 200 sentences from a corpus of essays written by non-native English speakers for a college-level English proficiency exam. In addition to sentences containing grammatical errors, we also deliberately sampled sentences that contained no grammatical errors in order to determine how our techniques perform in those cases. In total, 162 of the sentences contained at least one error, and the remaining 38 were perfectly grammatical. For both errorful as well as grammatical sentences, we sampled sentences of different lengths (under 10 words, 10-20 words, 20-30 words, 30-40 words, and over 40 words). The 162 errorful sentences varied in the number and type of errors present. Table 1 shows the distribution of the number of errors across these 162 sentences.

Specifically, the error types found in these sentences included prepositions, articles, punctuation, agreement, collocations, confused words, etc. Some only contained a handful of straightforward errors, such as "*In recent day, transportation is one of the most important thing to support human activity*", where *day* and *thing* should be pluralized. On the other hand, others were quite garbled to the point where it was difficult to understand the meaning, such as "*Sometimes reading a book is give me information about the knowledge of life so that I can prevent future happened but who knows that when it will happen and how fastly can react to that happen.*" During development, we noticed that the round-trip translation process could not handle misspelled words, so we manually corrected all spelling mistakes which did *not* result in a real word.[6]

[6] A total of 82 spelling errors were manually corrected.

5 Evaluation

In order to evaluate the six techniques for generating corrections, we designed an evaluation task where the annotators would be shown a correction along with the original sentence for which it was generated. Since there are 6 corrections for each of the 200 sentences, this yields a total of $1,200$ units for pairwise preference judgments. We chose two annotators, both native English speakers, each of whom annotated half of the judgment units.

Given the idiosyncrasies of the statistical machine translation process underlying our correction techniques, it is quite possible that:

- A correction may fix some, but not all, of the grammatical errors present in the original sentence, and
- A correction may be more fluent but might change the meaning of the original sentence.
- A correction may introduce a new disfluency, even though other errors in the sentence have been largely corrected. This is especially likely to be the case for longer sentences.

Therefore, the pairwise preference judgment task is non-trivial in that it expects the annotators to consider two dimensions: that of grammaticality and of meaning. To accommodate these considerations, we designed the evaluation task such that it asked the annotators to answer the following two questions:

1. **Grammaticality**. The annotators were asked to choose between three options: "*Original sentence sounds better*", "*Correction sounds better*" and "*Both sound about the same*".
2. **Meaning**. The annotators were asked to choose between two options: "*Correction preserves the original meaning*" and "*Correction changes the original meaning*". It should be noted that determining change in or preservation of meaning was treated as a very strict judgment. Subtle changes such as the omission of a determiner were deemed to change the meaning. In some cases, the original sentences were too garbled to determine the original meaning itself.

	$C > O$	$C = O$	$C < O$
Meaning = 1	S	D	F
Meaning = 0	F	F	F

Table 2: A matrix illustrating the Success-Failure-Draw evaluation criterion for the 162 errorful sentences. The rows represent the meaning dimension (1 = meaning preserved, 0 = meaning changed) and the columns represent the grammaticality dimension ($C > O$ denotes correction being more grammatical than the original, $C = O$ denotes they are about the same and $C < O$ denotes that the correction is worse). Such a matrix is computed for each of the six techniques.

5.1 Effectiveness

First, we concentrate our analysis on the original sentences which contain at least one grammatical error. We aggregated the results of the pairwise preference judgments for each of the six specific correction generation techniques and applied the strictest evaluation criterion by computing the following, for each technique:

- **Successes.** Only those sentences for which the correction generated by method is not only more grammatical but also preserves the meaning.

- **Failures.** All those sentences for which the correction is either less grammatical or changes the original meaning.

- **Draws.** Those sentences for which the correction preserves the meaning but sounds about the same as the original.

Table 2 shows a matrix of the six possible combinations of grammaticality and meaning for each method and demonstrates which cells of the matrix contribute to which of the above three measures: Successes (S), Failures (F) and Draws (D).

In addition to the six techniques, we also posit an oracle in order to determine the upper bound on the performance of our round-trip translation approach. The oracle picks the most accurate correction generation method for each individual sentence out of the 6 that are available. For sentences where none of the six techniques produce an adequate correction, the oracle just picks the original sentence. Table 3 shows how the various techniques (including the oracle) perform on the 162 errorful sentences as measured by this criterion. Based on this criterion, the greedy technique performs the best compared to the others since it has a higher success rate (36%) and a lower failure rate (31%). The oracle shows that 60% of the errorful sentences are fixed by at least one of the six correction generation techniques. We show examples of success and failure for the greedy technique in Figure 4.

5.2 Effect of sentence length

From our observations on development data (not part of the test set), we noticed that Google Translate, like most statistical machine translation systems, performs significantly better on shorter sentences. Therefore, we wanted to measure whether the successes for the best method were biased towards shorter sentences and the failures towards longer ones. To do so, we measured the mean and standard deviation of lengths of sentences comprising the successes and failures of the greedy technique. Successful sentences had an average length of approximately 18 words with a standard deviation of 9.5. Failed sentences had an average length of 23 words with a standard deviation of 12.31. These numbers indicate that although the failures are somewhat correlated with larger sentence length, there is no evidence of a significant length bias.

5.3 Effect on grammatical sentences

Finally, we also carried out the same Success-Failure-Draw analysis for the 38 sentences in our test set that were perfectly grammatical to begin with. The analysis differs from that of errorful sentences in one key aspect: since the sentences are already free of any grammatical errors, no correction can be grammatically better. Therefore, sentences for which the correction preserves the meaning and is not grammaticality worse will count as successes and all other cases will count as failures. There are no draws. Table 4 illustrates this difference and Table 5 presents the success and failure rates for all six methods. The greedy technique again performs the best out of all six methods and successfully retains the meaning and grammaticality for almost 80% of

Method	Success	Draw	Failure
Baseline LM (B)	21% (34)	9% (15)	70% (113)
Greedy (G)	36% (59)	33% (52)	31% (51)
1-best (1)	32% (52)	30% (48)	38% (62)
LM Re-ranked (L)	30% (48)	17% (27)	54% (87)
Product Re-ranked (P)	23% (37)	38% (61)	40% (64)
LM Composition (C)	19% (31)	12% (20)	69% (111)
Oracle	60% (97)	40% (65)	-

Table 3: The success, draw and failure rates for the six correction generation techniques and the oracle as computed for the 162 errorful sentences from the test set. The oracle picks the method that produces the most meaning-preserving and grammatical correction for each sentence. For sentences that have no adequate correction, it picks the original sentence. Numbers in parentheses represent counts.

Success	That's why I like to *make travel by* using my own car. **That's why I like to travel using my own car**.
	Having discuss all this I must say that I *must rather prefer* to be a leader than just a member. **After discussing all this, I must say that I would prefer to be a leader than a member.**
Failure	And simply *there* is fantastic for everyone **All magical and simply there is fantastic for all**
	I hope that *share a room with she can be certainly kindle*, because she *is likely me* and so *will not be problems with she*. **I hope that sharing a room with her can be certainly kindle, because it is likely that I and so there will be no problems with it.**

Figure 4: Two examples of success and failure for the Greedy (G) technique. Original sentences are shown first followed by the corrections in bold. Grammatical errors in the original sentences are in italics.

the grammatical sentences.[7]

	$C > O$	$C = O$	$C < O$
Meaning = 1	-	S	F
Meaning = 0	-	F	F

Table 4: A matrix illustrating the Success-Draw-Failure evaluation criterion for the 38 grammatical sentences. There are no draws and sentences for which corrections preserve meaning and are not grammatically worse count as successes. The rest are failures.

Method	Success	Failure
Baseline LM (B)	26% (10)	74% (28)
Greedy (G)	79% (30)	21% (8)
1-best (1)	61% (23)	39% (15)
LM Re-ranked (L)	34% (13)	66% (25)
Product Re-ranked (P)	42% (16)	58% (22)
LM Composition (C)	29% (11)	71% (25)

Table 5: The success and failure rates for the six correction generation techniques as computed for the 38 grammatical sentences from the test set.

6 Discussion & Future Work

In this paper, we explored the potential of a novel technique based on round-trip machine translation for the more ambitious and realistic task of whole-sentence grammatical error correction. Although the idea of round-trip machine translation (via a single pivot language) has been explored before in the context of just preposition errors, we expanded on it significantly by combining multiple round-trip translations and developed several new methods for producing whole-sentence error corrections. Our oracle experiments show that the ideas we explore have the potential to produce whole-sentence corrections for a variety of sentences though there is clearly room for improvement.

An important point needs to be made regarding the motivation for the round-trip translation approach. We claim that this approach is useful not just because it can produce alternative renderings of a given sentence but primarily because each of those

[7] An oracle for this setup is uninteresting since it will simply return the original sentence for every sentence.

renderings is likely to retain at least some of meaning of the original sentence.

Most of the problems with our techniques arise due to the introduction of new errors by Google Translate. One could use an error detection system (or a human) to explicitly identify spans containing grammatical errors and constrain the SMT system to translate only these errorful spans while still retaining the rest of the words in the sentence. This approach should minimize the introduction of new errors. Note that Google Translate does not currently provide a way to perform such **selective translation**. However, other open-source SMT systems such as Moses[8] and Joshua[9] do. Furthermore, it might also be useful to exploit n-best translation outputs instead of just relying on the 1-best as we currently do.

As an alternative to selective translation, one could simply extract the identified errorful spans and round-trip translate each of them individually. For example, consider the sentence: "*Most of all, luck is* null prep *no use without* a *hard work.*" where the preposition *of* is omitted and there is an extraneous article *a* before "hard work". With this approach, one would simply provide Google Translate with the two phrasal spans containing the errors, instead of the entire sentence.

More generally, although we use Google Translate for this pilot study due to its easy availability, it might be more practical and useful to rely on an in-house SMT system that trades-off translation quality for additional features.

We also found that the language-model based techniques performed quite poorly compared to the other techniques. We suspect that this is due to the fact that Google Translate already employs large-order language models trained on trillions of words. Using lower-order models trained on much smaller corpora might simply introduce noise. However, a detailed analysis is certainly warranted.

In conclusion, we claim that our preliminary exploration of large-scale round-trip translation based techniques yielded fairly reasonable results. However, more importantly, it makes it clear that, with additional research, these techniques have the potential to be very effective at whole-sentence grammatical error correction.

[8] `http://www.statmt.org/moses`

[9] `https://github.com/joshua-decoder`

Acknowledgments

We would like to thank Aoife Cahill, Michael Heilman and the three anonymous reviewers for their useful comments and suggestions. We would also like to thank Melissa Lopez and Matthew Mulholland for helping with the annotation.

References

Ion Androutsopoulos and Prodromos Malakasiotis. 2010. A Survey of Paraphrasing and Textual Entailment Methods. *J. Artif. Int. Res.*, 38(1):135–187.

Srinivas Bangalore, German Bordel, and Giuseppe Riccardi. 2001. Computing Consensus Translation from Multiple Machine Translation Systems. In *Proceedings of ASRU*, pages 351–354.

Regina Barzilay and Lillian Lee. 2003. Learning to Paraphrase: An Unsupervised Approach Using Multiple-Sequence Alignment. In *Proceedings of HLT-NAACL 2003*, pages 16–23.

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.

Yang Feng, Yang Liu, Haitao Mi, Qun Liu, and Yajuan Lü. 2009. Lattice-based System Combination for Statistical Machine Translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3*, pages 1105–1113.

Matthieu Hermet and Alain Désilets. 2009. Using First and Second Language Models to Correct Preposition Errors in Second Language Authoring. In *Proceedings of the Fourth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 64–72.

Claudia Leacock, Martin Chodorow, Michael Gamon, and Joel Tetreault. 2010. *Automated Grammatical Error Detection for Language Learners*. Synthesis Lectures on Human Language Technologies. Morgan Claypool.

Nitin Madnani and Bonnie J. Dorr. 2010. Generating Phrasal and Sentential Paraphrases: A Survey of Data-driven Methods. *Computational Linguistics*, 36(3).

Bo Pang, Kevin Knight, and Daniel Marcu. 2003. Syntax-based Alignment of Multiple Translations: Extracting Paraphrases and Generating New Sentences. In *Proceedings of HLT-NAACL*, pages 102–109.

Y. Albert Park and Roger Levy. 2011. Automated Whole Sentence Grammar Correction using a Noisy Channel Model. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, pages 934–944.

Antti-Veikko Rosti, Necip Fazil Ayan, Bing Xiang, Spyros Matsoukas, Richard Schwartz, and Bonnie Dorr. 2007. Combining Outputs from Multiple Machine Translation Systems. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 228–235.

Matthew Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. 2009. Fluency, Adequacy, or HTER? Exploring Different Human Judgments with a Tunable MT Metric. In *Proceedings of the Fourth Workshop on Statistical Machine Translation at the 12th Meeting of the European Chapter of the Association for Computational Linguistics (EACL-2009)*.

Shiqi Zhao, Haifeng Wang, Xiang Lan, and Ting Liu. 2010. Leveraging Multiple MT Engines for Paraphrase Generation. In *COLING*, pages 1326–1334.