Essay Scoring with Grammatical Error Detection

Brandon Ross Katz, Robert Plant, Nina Kuklisova
W 266: Natural Language Processing
UC Berkeley School of Information
{brosskatz, drewplant, nkuklisova}@ischool.berkeley.edu

Abstract

English proficiency is an important skill in the world today. For non-native English speakers, an automated system capable of reviewing practice essays would assist English learners without requiring involvement of native essay graders. This project explored the use of an LSTM-RNN machine model for identifying existence of grammatical errors in essays written by non-native learners of English. The relatively high accuracy obtained suggests that using LSTM-RNN models is a promising approach and recommends future opportunities for refinements.

1 Introduction

In an age of increasing globalization, acquisition of English communication skills is a priority for people worldwide. For people where English is not the primary language used in school, developing English fluency requires lots of practice. Given that many *English as a Second Language* (ESL) students are lucky if they encounter a single native speaker before they reach college, having practice essays reviewed and corrected by a native speaker is an unfortunately rare occurrence.

Automated grammatical error detection could assist English learners in advancing their written and spoken communication skills without reliance on native-speaking graders. This paper discusses an automatic grammatical error tagger, using a *Long Short Term Memory Recurrent Neural Network* (LSTM-RNN.) Features for training the LSTM-RNN included GLoVe word embeddings and parts-of-speech. Future work will explore utilization of a bi-directional LSTM and n-gram features.

The goal of this system is to identify the location of errors in students' written essays without categorizing the type of error. The system achieved 18.1% precision, 60.1% recall, and 27.8% F2-score. These results were achieved despite having a small training corpus size, (30,000 tokens with approximately 5% of the tokens being an error), inconsistency between labelers, and unclear error-to-word assignment without knowing the intent of the essay author.

2 Project Overview

2.1 Datasets used

We initially explored the First Certificate in English (FCE) exams dataset (`http://ilexir.co.uk/applications/clc-fce-dataset/`) but this dataset was found to contain inconsistent parsing syntax and so we switched to using the dataset from the 2013 Conference on Computational Natural Language Learning (CoNLL) (`http://www.comp.nus.edu.sg/~nlp/conll13st.html`) Shared Task on Grammatical Error Corrections.

Global Vectors for Word Representation, (GLoVe) embeddings were taken from a pre-trained dataset (`http://nlp.stanford.edu/projects/glove/`).

2.2 Background and problem approach

The shared task from the 2013 CoNLL involved prediction of types of grammatical errors from a corpus of essays written by non-native English speakers. [2].

This dataset focused on 5 grammatical errors occurring most frequently among non-native English learners:

Articles or determiner errors (ArtOrDet)
Noun number (Nn)
Prepositions (Prep)
Verb form (Vform)
Subject-verb agreement (SVA)

Additionally we added a *misspelling* category not originally included in the CoNLL'13 in order to assist with error-training.

The breakdown for errors in our dataset is shown below (Figure 1):

The highest accuracy for detecting errors for the conference [3] was achieved using Averaged Perceptron and Naive Bayes classifiers trained on the Google Web IT 5-gram corpus. Other teams tried similar approaches with various outcomes; all of the 5 top-scoring methods additionally used parts-of-speech as a feature.

Our methodology was based on an LSTM in order to account for context words for learning error identification.

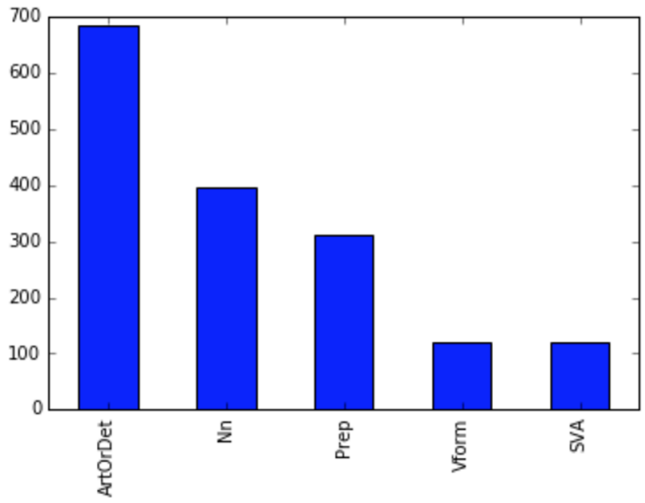

Fig. 1. Error count by category

In addition, LSTM RNNs have been shown in the literature to lessen the burden of designing elaborate feature sets. [5]

2.3 Parsing tools used

First, our dataset was XML-parsed and divided into sentences to feed into the model. The CoNLL dataset was split into training and test data (80% and 20% of the sentences respectively).

Mistake annotations in the essay corpus were converted to error / non-error boolean values. While we focused on error *existence*, we believe that an LSTM model could eventually learn (given sufficient data) error types (such as Subject-Verb Agreement) by naturally developing dependency features.

2.4 Part-of-Speech-Tagging

Part of Speech Tagging was accomplished using TextBlob's Perceptron Tagger (`http://stevenloria.com/tutorial-state-of-the-art-part-of-speech-tagging-in-textblob/`). Because several of the error categories were based on the part-of-speech of the word in question, we believed that Parts of Speech would be very helpful for better modeling both location and type of error.

2.5 GLoVe Embeddings

Research has demonstrated that GLoVe embeddings encode latent relationships between words in a multi-dimensional vector that can be incorporated as features of a language model. After an extensive class experience with the Global Vectors for Words Representations (GloVe) Model [1], we attempted to train GLoVe embeddings on the first 30,000,000 words of British National Corpus (BNC). The BNC was chosen because our language learners from Singapore utilized British English in their essays. Unfortunately, these embeddings did not encode relationships as rich as the pre-trained embeddings available to the public. Therefore, the team decided to utilize pre-trained GloVe embeddings.

2.6 Synthetic Data

To compensate for the lack of data in the CoNLL Shared Task corpus, we attempted to reproduce synthetic errors for the error types on which the LSTM performed worst: Verb Form errors and Preposition Usage errors. For the Verb Form errors, our methodology was to identify the verb in each sentence of the Brown Corpus, modify the form of that error, and leave the rest of the sentence unchanged. We could then tag every word in that sentence as correct with the exception of the modified verb.

Similarly, we took another set of sentences and identified common prepositions in the sentence. We then replaced those prepositions with another common preposition, mimicking an error that an English student might make. Once again, we tagged only the modified word as an error and accepted the other words as correct.

Finally, we experimented with adding clean sentences from the Brown Corpus into the training data but leaving the test dataset unchanged. This was meant to test the hypothesis that adding clean English data might help the model to discriminate between correct and incorrect English usage across a variety of grammatical constructs.

We experimented with adding 200, 500, and 1000 sentences from each group to the dataset (against 1,100 sentences in our training dataset, and none of these levels improved test performance. We also experimented with adding only error-sentences and only clean sentences at various levels. Once again, adding any level of synthetic data did not improve test performance. At best, performance was close to its original level without synthetic data. We hypothesize that another methodology for creating synthetic data, for example by being more specific about the types of Verb Form and Prepositional errors students create, would lead to improvements in test dataset results. However, it might be more reasonable to simply collect more training data.

3 Model

3.1 Baseline Model

Using these, we built a baseline model, using a 2-level LSTM RNN. For each input word, this model estimates its probability of being an error or not. A diagram of our model is in figure 2. After setting the correction threshold to 0.1 probability with test data, we found our precision to be 0.14, recall 0.47, F2-score 0.23. When raising the threshold up to .2, Test Data precision rose to .31, recall falls slightly to .41, and F2-Accuracy reaches its peak of .35.

3.2 Prospective Use of Bi-Directional LSTM-RNN

We believe that scores will be improved by converting from a one-directional LSTM to a bi-directional LSTM. This will allow the model to account for dependencies occurring both before and after the error while doubling the number of feature-generating feedback loops built into the system. We

are attempting to deploy this bi-directional LSTM before the end of the class period, but may postpone this portion for future research.

3.3 Prospective Use of N-Gram Features

Several of the winning models in the 2013 CoNLL Shared Task utilized N-Gram features. Intuitively, this would capture situations where, for example, plural nouns are used in conjunction with singular verbs. This would be a rare, almost non-existent, N-Gram and would signal the model that there may be an error somewhere within that word's last N-words. This will be saved for future research.

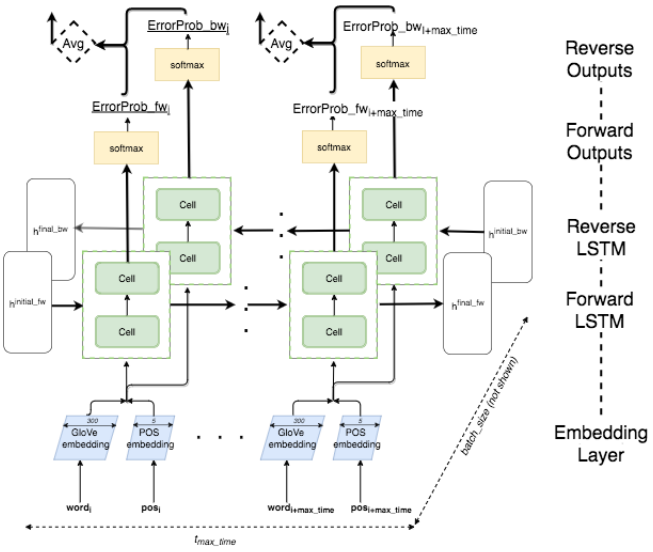

Fig. 2. Model Diagram

4 Results

We generated initial features for the dataset, including Part of Speech Tag, Previous Tag, Next Tag, Previous Word, Next Word. Yet, as we can see from the results table in Figure 3, our LSTM still couldn't identify all the errors in essays encountered.

4.1 Accuracy with Different Error Types

Among the different error types, this model was most successful with Noun Counts, followed by Subject-Verb Agreement, Prepositions, and Articles or Determinants. Verb Form was by far the least accurate error type. We hypothesize that Noun Counts were easiest to recognize because students were using similar nouns in response to the prompt and responding with similar errors regarding those nouns. The Article Errors were almost all caught at low thresholds, but the model is less confident of these errors (lower probability distribution across the entire model). Subject-Verb Agreement and Verb Form errors were difficult to distinguish, but

	Training Data			
Milestone	**Loss**	**Precision**	**Recall**	**F2**
Baseline	0.162	0.233	0.734	0.353
+ GloVe	0.152	0.270	0.755	0.397
+ GloVe + POS	0.142	0.279	0.800	0.414

	Test Data				
Milestone	**Loss**	**Precision**	**Recall**	**F2**	**AUC**
Baseline	0.169	0.164	0.583	0.257	0.8
+ GloVe	0.168	0.178	0.579	0.272	0.81
+ GloVe + POS	0.165	0.181	0.601	0.278	.0.82

Fig. 3. Results

it is notable that the model had such drastically different accuracies with these two error types even though they occur with the same part of speech.

4.2 Discussion of Errors

Error analysis enables us to determine what "assumptions" our model might make that lead to misclassified or misidentified errors. Here are a few examples of errors that we missed:

(ArtOrDet): human make influence the lifespan of people . according to < error > news , the police force had mobilized thousand of cops

(Vform): in the medical area and people 's living standard has < error > been improved . we cannot imagine our every move is

(Prep): as people felt in the past . since not all < error > of the human being is going to be people in the law sector . an example will be < error > in korea . in economic , we can use gdp low , it shows that most of people are living < error > in a hard life , life quality is bad , . before robots can fully replace labor , the decreasing < error > of generating power indicate less creativity and less goods available

(Nn): on the other hand , in terms of tangible < error > aspect mainly in the economic drive and desire of the

(SVA): being is going to be a doctor , people < error > relies the skills of treatment of doctors and hence can fully replace labor , the decreasing of generating power < error > indicate less creativity and less goods available for us to

(Spelling): < error >the modification was so obvious that people can easily recognize frequency identification can be used to track people . < error > the air cargo of the valujet plane was on fire

There are a few conclusions we can draw from these misclassified words:

1. Many of the errors would be easily caught by a model

that could somehow proxy what "sounds right" in the English language. This indicates that a feature which utilizes n-grams from typical English would probably outperform our current model. This is because we miss subject-verb agreements, preposition usage, and count vs. non-count noun issues that would almost never occur in everyday usage of the language (i.e. "anything are fault", "people relies", "issue of implant", "terrible society problem", "about our future generation","considering for human safety"). We could also use something like edit-distance on student n-grams to identify possible corrections that are closest to what the language learner is trying to say.

2. Additional errors might be gleaned by feeding the model more error-free, native English language so that its understanding of dependencies could be improved, such as "types of are" rather than "types of is". However, attempts to do this using the Brown corpus were unsuccessful. Ideally, we would choose error-free, native English language that mimics closely the subject matter and word usage of the English language learners.

3. Better classification of errors might occur if we created synthetic data from native English language that gave the model more passes at similar types of errors, such as plural noun should not get singular verb (although we would want to see whether these would be caught by incorporating n-grams into the model). Once again, this was attempted with the Brown corpus, but a lack of improvement could be attributed either to the corpus used and its differences from student essays or to the methodology of synthetic data creation applied.

4. We could also consider incorporating singluar vs. plural verb tags and subject tags so that the model could start to expect when a singular noun and verb might require agreement and where the noun represents the subject. This would also be aided by features generated by a dependency parse.

5. Lastly, rather than using individual words as errors tagged in the corpus, we could divide the corpus into trigrams around each error (error -1, error word, error +1) so that the model can more easily identify a cluster of words that have a grammatical error in their relationship to each other and the rest of the essay rather than a single word that has an error because of where it is relative to other words. However, converting our model to bi-directional may also improve its ability to accurately assign which word in a phrase should receive an error tag, since today it can only tag the next word as an error based on the word it saw previously, whereas a human annotator can choose any word available.

5 Comparison of Results

In CoNLL'13, the winning team achieved a 46% precision, 23% recall and 31% F1-score; however, these scores involved categorizing errors, not just locating them. A future iteration of this project would combine LSTM error probabilities with the features used to create the LSTM in a Naive Bayes or other simple classification model. In most cases, the error is simply the part-of-speech of the error in that location, but a model that assigns a tag probability to a particular word would add another level of filtering that removes tagged errors associated with parts of speech that have never been seen before.

6 Conclusion

This project demonstrated viability of an LSTM-RNN model for identifying location of grammatical errors in essays written by English language learners. Features included Parts of Speech, GLoVe embeddings, and the features generated by the RNN. Future iterations of the model would incorporate a bi-directional RNN to account for features before and after each word as well as N-Gram features that capture commonly collocated words that indicate correct usage. English language instruction systems may be able to combine smart pedagogical methods with adequately accurate models to enable learner self-correction. In fact a slightly-inaccurate model might motivate learners to critically question the feedback they receive.

Another technique for future consideration for additional accuracy might be the attention-based encoder-decoder model, as described by Schmaltz et al. [4]. However, this model is significantly more complex. Alternative tools which could also improve the results are statistical tools such as Actual Parse Probability / Estimated Parse Probability method or the Treebank method. An excellent comparison of these and other methods has been done by Wagner [6].

For future Shared Tasks, annotator consistency seems imperative. As noted in CoNLL'14 [7], different graders score the same essays in a different way. One example that the CoNLL'14 organizers note is an instance where the accuracy or error detection is 70% for one grader, but only 28% for the other grader. Therefore, there scores would be probably different when other annotator's work was used as reference. This level of disagreement makes the extraction of any underlying pattern that an algorithm might recognize near impossible. Some methodology for evaluating the quality of training data prior to training might be useful in ensuring that Task participants are working toward an achievable goal.

Finally, a challenge for members of the education technology development community is to create systems that enable rapid, consistent annotation of data such that models can be continually improved as more data is collected. We expect methods developed in this and other papers from the Shared task to be more successful in an environment that facilitates larger quantities of clean data being collected, annotated, and manipulated.

References

[1] J. Pennington, R. Socher, C. D. Manning, *GloVe: Global Vectors for Word Representation*; Computer Science Department, Stanford University

[2] H. T. Ng, S. M. Wu, Y. Wu, C. Hadiwinoto, J. Tetreault, *The CoNLL-2013 Shared Task on Grammatical Error Correction*; Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, pages 1 - 12, Sofia, Bulgaria, August 8-9, 2013

[3] A. Rozovskaya, K.-W. Chang, M. Sammons, D. Roth, *The Univeristy of Illinois System in ConLL-1013 Shared Task*

[4] A. Schmaltz, Y. Kim, A. M. Rush, S. Shieber, *Sentence-Level Grammatical Error Identification as Sequence-to-Sequence Correction*

[5] D. Chen, C. D. Manning, *A fast and accurate dependency parser using neural networks*; Empirical Methods in Natural Language Processing *EMNLP*.

[6] J. Wagner, *Detecting Grammatical Errors with Treebank-Induced, Probabilistic Parsers* A dissertation submitted in partial fulfillment of the requirements for the award of Doctor of Philosophy to the Dublin City University School of Computing

[7] H. T. Ng, S. M. Wu, T. Briscoe, C. Hadiwinoto, R. H. Susanto, C. Bryant, *The CoNLL-2014 Shared Task on Grammatical Error Correction*; Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 1 - 14, Baltimore, Maryland, 26-27 July 2014