TextBooster: Substituting Desired Vocabulary into Daily Reading Material*

Carrie Cai

Abstract

Studying for standardized exams (e.g. SAT, GRE) requires regular practice with advanced vocabulary, yet it is uncommon to encounter these difficult words in context because their occurrence in daily language is rare. We introduce the idea of *textboosting*: modifying regularly encountered sentences during web browsing so that they include desired vocabulary, while preserving the overall flow and meaning of the original sentence. To do this, we model appropriate word usage by training classifiers that predict contextual preferences of each word, using distant relatives of words to curate negative example sentences. Our experiments show that the use of ngram probability features enhances training performance on rare words, and that our model outperforms baseline on real substitutions despite limited training data. These results provide insight into how daily content could be transformed to encourage in-context learning of rare words.

*This document has been adapted from the instructions for earlier ACL and NAACL proceedings, including those for NAACL HLT15 by Matt Post and Adam Lopez, NAACL HLT12 by Nizar Habash and William Schuler, NAACL HLT10 by Claudia Leacock and Richard Wicentowski, NAACL HLT09 by Joakim Nivre and Noah Smith, for ACL05 by Hwee Tou Ng and Kemal Oflazer, for ACL02 by Eugene Charniak and Dekang Lin, and earlier ACL and EACL formats. Those versions were written by several people, including John Chen, Henry S. Thompson and Donald Walker. Additional elements were taken from the formatting instructions of the *International Joint Conference on Artificial Intelligence* and the *Conference on Computer Vision and Pattern Recognition*.

1 Introduction

Every year, millions of students need to study vocabulary in preparation for standardized exams such as the SAT or GRE. Despite limited time available to study, web browsing occurs frequently and contains rich opportunities for in-context learning. However, encountering advanced vocabulary is uncommon because their occurrence in daily language is rare. We introduce the notion of *textboosting*: modifying existing sentences to include desired vocabulary. Our goal is to make word substitutions while preserving the overall flow and meaning of the original sentence.

In this paper, we explore methods for identifying contexts when a word from a desired list can be reasonably substituted. First, we observe that most standardized exam vocabulary have rare occurrences, but approximately 10% of those words have close synonyms which appear frequently. We leverage these observations to select a subset of vocabulary words that are more appropriate for substitution. We then conducted a feasibility analysis, and found that as many as 23 substitutions could be made in the top seven front-page articles of the New York Times on a given day, an average of 3 substitutions per article.

Given a sentence and a target word for substitution, our algorithm determines if the target word should in fact be inserted. For instance, given the sentence: "The infusion dials back the *tendency* of the recipient's immune system to attack the transplant," we automatically determine if the target word "propensity" can be substituted. Because synonyms cannot be substituted in any context (Edmonds and

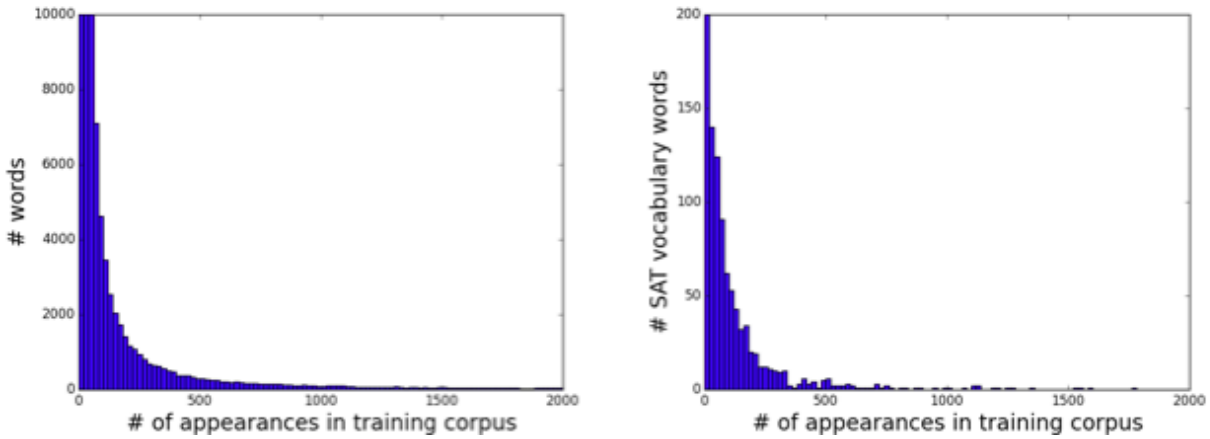

Figure 1: Frequency of words in in the 4 million sentence training corpus. a) Frequency distribution of all words in the training corpus. b) Frequency distribution of SAT words: the average number of appearances is 159 (σ=385), with 89% of vocabulary words appearing fewer than 300 times.

Hirst, 2002), our algorithm predicts whether the target word can be appropriately inserted by analyzing its distributional properties in existing bodies of text. Sentences that pass this test are rewritten using the target vocabulary word. Our experiments show that the use of ngram probability features enhances training performance on rare words, and that our model achieves reasonable performance (75.9% precision, 64.1% recall) on real substitutions despite limited training data.

2 Related Work

Based on growing evidence that regular and spaced exposure to vocabulary result in greater learning gains (Dempster, 1987), recent work on *microlearning* (Gassler et al., 2004) has explored a variety of ways to distribute learning into small units throughout a person's daily life. Existing approaches aim to increase learning opportunities by finding text appropriate to the learner's reading level (Heilman et al., 2008), translating encountered words on the fly to a foreign language for second language learning (Cai et al., 2015; Trusty and Truong, 2011), or suggesting vocabulary words relevant to a person's location (Edge et al., 2011). Beyond suggesting vocabulary relevant to the user, we instead *modify* existing content a person encounters to include desired learning content. To our knowledge, this work is the first to actively transform content so that it becomes relevant to the user, even if it was not initially.

Furthermore, while an existing body of work has aimed to automatically generate paraphrases given a sentence (Barzilay and Lee, 2003; Kauchak and Barzilay, 2006; Pang et al., 2003), our work differs from this work in goal and methodology. First, our aim is not to produce any paraphrase, but rather one that substitutes in a desired word or piece of text within a set. As such, these desired words also tend to be rare because they represent more advanced vocabulary. Second, because our goal is to inspire opportunistic learning of vocabulary, substitution is not mandatory for all words. Hence, a core challenge of this work is in determining which vocabulary words to select for learning, and which candidate words to substitute.

3 Methods

The input to our method consists of a sentence $S = s_1...s_n$, a position in the sentence i, and a target word w. Our goal is to predict whether w can occur at position i within this context.

3.1 Data

We use the 2012 English Gigaword Corpus for sentence-level data. The English Gigaword Corpus contains data from a variety of news sources, which maps well to the kinds of news articles a typical person might encounter online. For target words, we downloaded a set of 1000 SAT vocabulary words

from quizlet.com, an online platform for educational flashcards.

3.2 Candidate Selection

We first observe that SAT words tend to appear infrequently in daily language. Among the 4 million sentences available for training, the average SAT vocabulary word appears only 159 times (σ=385) (Figure 1b). This supports the notion that typical words required for study are encountered infrequently, motivating a need for textboosting. However, sparse training data also raises two challenges: 1) **sparse data**: low training data makes it difficult to learn word contexts accurately, and 2) **sparse context**: some words may simply appear in highly specific contexts (e.g. "agriculture"), making them less appropriate for text boosting by substitution.

To address these challenges, we identify a subset of words that may be more reasonable candidates for textboosting. First, we focus only on the subset of words with at least 150 training examples. To mitigate the sparse data problem, we select words that have *close relatives* so that the likelihood of appropriate substitution is high. For each target word w, we identify a different word c to be a close relative to w if c is in the thesaurus for w, and the word2vec similarity score between w and c is at least 0.45. By combining synonym knowledge with word2vec similarity, we hope to ensure that close relatives are both semantically similar to w and appear in similar contexts as w. We used a thesaurus API[1] instead of WordNet because we found that the synonyms provided were more comprehensive than those in WordNet.

Second, to mitigate the sparse context problem, we keep only words with frequently occurring close relatives. Despite infrequent appearances, some vocabulary words have meanings that do appear frequently in daily language. For example, *gratuitous* appears only 267 times in the training corpus, but its close relative *unnecessary* appears 1713 times. We thus identify 70 vocabulary words whose close relatives have a combined frequency of at least 500. We use 500 as a threshold because it is beyond the knee of the frequency curve shown in Figure 1b). Although we focus on 70 words in this paper, the above constraints could be relaxed or adjusted depending on the number of vocabulary words desired for learning.

Lastly, we conduct a feasibility analysis on the 70 words selected. To do this, we scraped the top seven front-page articles on the New York Times on December 7, 2015. Given the set of close neighbors identified for each target word, we extracted sentences containing those close neighbors, and manually identified which instances could be appropriately replaced with the target word. We found that 23 substitutions could be made among these top seven front-page articles, an average of 3 substitutions per article. Given that learning requires regular exposure to target vocabulary, these preliminary findings provide support for the viability of textboosting for education.

3.3 Contextual Substitution

Next, we determine for each candidate pair (s_i, w_j) whether w_j is a valid substitution for c_i in the context of $(s_1 ... s_{i-1} \square s_{i+1} ... s_n)$. We formulate contextual substitution as a binary classification task. For each word, we train a classifier that predicts contextual preferences of w_j, similar to methods used in supervised word sense disambiguation and paraphrasing (Kauchak and Barzilay, 2006).

TextBooster ought to accurately identify positive contexts for the target word without missing too many opportunities for substitution. In the absence of manually annotated training data, we modify the strategy described in (Kauchak and Barzilay, 2006) to automatically create a training corpus. For each word, we produce an equal number of positive and negative instances. Positive instances are produced by collecting sentences that contain the word w. Negative instances are produced by collecting sentences that contain *distant relatives* of w (Figure 2). Distant relatives are words that may have some similarity to w, but cannot be used interchangeably as w. We use distant relatives rather than random instances as negative examples, so that the model can learn more nuanced contexts for appropriate usage of w. We define distant relatives as words that are two degrees of separation away from the target word: for each target word w and one of its close relatives c, a distant relative d appears in the thesaurus of c, but not in the thesaurus of w. To further ensure that w

[1] https://words.bighugelabs.com/

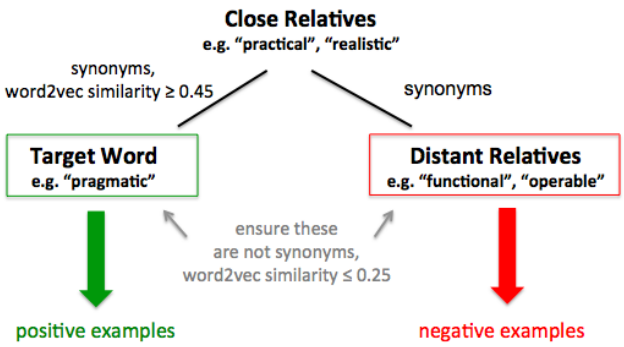

Figure 2: For each word, positive training examples consisted of sentences containing the target word, and negative examples consisted of sentences containing distant relatives of the target word. Distant relatives were two degrees of separation from the target word by synonym lookup, and had a low word2vec similarity score. Close relatives were synonyms of the target word with a high word2vec similarity score.

and d are sufficiently different in meaning, we require their word2vec similarity score to be at most 0.25.

We represent context using n-grams surrounding (but excluding) the target word. In negative examples, the distant relative serves as the target word. Given position i of a target word, we extract all n-grams ($n = 1...4$) beginning at position $i-3$ and ending at position $i+3$. We include both word n-grams and part of speech n-grams.

Once classifiers have been trained, we find candidate sentences for substitution by identifying sentences containing a close neighbor of each word w in our vocabulary set. Removing the close neighbor from the sentence, we apply the classifier for w to the sentence context $(s_1...s_{i-1}\square s_{i+1}...s_n)$. If the classifier yields a positive prediction, we rewrite the sentence as $(s_1...s_{i-1}ws_{i+1}...s_n)$.

One implication of this method is the need to train a large number of classifiers, one for each vocabulary word. In practice, however, this training can be completed beforehand and entirely offline, such that the contextual substitution itself can be done instantaneously. Given that learners may seek to learn only a manageable set of vocabulary at a time in a repeated manner, the number of potential candidates is also well-contained.

features	accuracy	precision	recall	F1
ngrams from i-1 to i+1	0.707	0.741	0.673	0.694
+ ngrams from i-3 to i+3	0.735	0.751	0.725	0.734
+ POS from i-3 to i+3	0.758	0.766	0.750	0.756
+ $P(w_i \mid w_{i-2}, w_{i-1})$	0.776	0.786	0.768	0.774

Table 1: Training performance of word-based classifiers by feature. The trigram probability feature (last row) increased performance for rare words with less than 300 training sentences. For each word, positive examples consisted of sentences containing the word, and negative examples consisted of sentences containing distant relatives of the target word.

4 Evaluation

4.1 Evaluation of Features

We first present training results with respect to the features described above (Table 1). We find that expanding the ngram context surrounding the target word from $(i-1, i+1)$ to $(i-3, i+3)$ substantially increases recall by 6%. Including part of speech features further increases accuracy by 2% (75.8% accuracy).

Lastly, we find that many of the lowest performing words have very little training data. Thus, for words with fewer than 300 training examples, we include an additional trigram probability feature: given a word at index i in a sentence, we retrieve the trigram conditional probability $P(w_i|w_{i-2}, w_{i-1})$ from the Google 1 Terabyte 5Gram corpus. The feature encodes whether or not the trigram probability is greater than a threshold. We found that 0.00001 was a reasonable cut-off. This additional feature increases overall accuracy by 2%, yielding 79% precision and 77% recall.

4.2 Evaluation of Substitution Quality

In the previous section, we observed that simple features yield reasonable performance when predicting the context of appropriate word usage, using distant relatives as negative examples in the absence

model	accuracy	precision	recall	F1
random	0.502	0.723	0.494	0.586
TextBooster	0.591	0.759	0.641	0.695

Table 2: Performance of TextBooster and a random baseline on 100 sentences. Ground truth appropriateness of substitutions were obtained by averaging the ratings of two human annotators.

of ground truth human labels. In this section, we perform actual substitutions using our pre-trained classifiers, and evaluate the quality of these substitutions.

To perform substitution, we first gather all sentences in our test set containing close neighbors of the 70 target words. We then randomly select 100 of these sentences for classification and human evaluation. Given each sentence $(s_1 ... s_{i-1} c s_{i+1} ... s_n)$ and close neighbor c, we apply our pre-trained classifier to the sentence to determine whether or not to substitute target word w for c. We also apply a baseline method that randomly assigns binary classifications to each sentence.

Two human annotators then rated each sentence, both of whom were native English spearkers. So that we could evaluate both precision and recall, the sentences presented to annotators were always substituted with their respective target words, regardless of the classifier output. The annotators were asked to indicate the extent to which "The highlighted word is used appropriately in this sentence," on a scale from 1 (Strongly disagree) to 4 (Strongly agree), or "can not tell without more information." They were told that some sentences may have words substituted, but were not told which sentences. For each sentence, we average the two ratings, and consider average ratings greater than 2.5 to be a positive classification.

We find that our model achieves 75.9% precision and 59.1% recall (Table 2). In comparison to the baseline model (72.3% precision, 49.4% recall), our model performs slightly higher (3.1%) on precision and substantially higher (8.6%) on recall. In addition, a majority of the sentences were annotated by humans as being appropriate (72.7%), suggesting that the process by which we identified candidate target words and close neighbors (see Section 3.2 above) was reasonable.

5 Conclusion and Future Work

This paper introduces the idea of substituting desired vocabulary into existing sentences for increased educational exposure, and presents a process for doing so with reasonable results. Our experiments suggest that our model can reasonably identify appropriate substitutions, with greater performance than an uninformed baseline, particularly on classification recall.

In developing the model, we also introduce a procedure for identifying examples of word usage in the absence of labeled training data, using distant relatives as negative examples. We further identified several ways to mitigate the effects of vocabulary sparsity, including target word selection, close neighbor selection, and the addition of language model probability features for rare words. We believe these challenges reflect hurdles that a real text-boosting system might face.

Our ultimate goal is to develop a method for unobtrusively introducing any desired educational content into existing text that is regularly encountered. Although our current method only implements substitutions, and operate at the word level, future work may explore not only substitutions but also the *addition* of text to existing content, such as adjectives and subordinate clauses. We believe our work opens up a new space of applications that actively modify regularly encountered text with desired content.

References

Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: an unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 16–23. Association for Computational Linguistics.

Carrie J Cai, Philip J Guo, James R Glass, and Robert C Miller. 2015. Wait-learning: Leveraging wait time for second language education. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3701–3710. ACM.

Frank N Dempster. 1987. Effects of variable encoding and spaced presentations on vocabulary learning. *Journal of Educational Psychology*, 79(2):162.

Darren Edge, Elly Searle, Kevin Chiu, Jing Zhao, and James A Landay. 2011. Micromandarin: mobile language learning in context. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3169–3178. ACM.

Philip Edmonds and Graeme Hirst. 2002. Near-synonymy and lexical choice. *Computational linguistics*, 28(2):105–144.

Gerhard Gassler, Theo Hug, and Christian Glahn. 2004. Integrated micro learning–an outline of the basic method and first results. *Interactive Computer Aided Learning*, 4.

Michael Heilman, Le Zhao, Juan Pino, and Maxine Eskenazi. 2008. Retrieval of reading materials for vocabulary and reading practice. In *Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications*, pages 80–88. Association for Computational Linguistics.

David Kauchak and Regina Barzilay. 2006. Paraphrasing for automatic evaluation. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 455–462. Association for Computational Linguistics.

Bo Pang, Kevin Knight, and Daniel Marcu. 2003. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 102–109. Association for Computational Linguistics.

Andrew Trusty and Khai N Truong. 2011. Augmenting the web for second language vocabulary learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3179–3188. ACM.