Assessing the Effectiveness of Corpus-Based Methods in Solving SAT Sentence Completion Questions

Eugene Tang*
Department of Computer Science, Princeton University, Princeton, USA.

* Corresponding author. Tel.: 609-986-8814; email: eugene.c.tang@gmail.com
Manuscript submitted June 17, 2015; accepted September 5, 2015.
doi: 10.17706/jcp.11.4.266-279

Abstract: SAT sentence completion questions are designed to assess knowledge of the English language. The ability to answer such types of questions has wide implications in optical character recognition, speech recognition, and word-suggestion programs. In our study, we analyze several statistical corpus-based methods through which to answer such questions, including normalized pointwise mutual information, co-occurrence frequencies, latent semantic analysis, and the word2vec neural net implementations of continuous bag of words (CBOW) and continuous skip-gram (CSKIP) models. We find that the co-occurrence frequency method has a strong performance with 52% correctness and that combining the co-occurrence frequency method with CBOW and CSKIP results in a state-of-the-art performance of 59%. The results of this study demonstrate that local context is a fairly strong measure in determining how well a word fits in a sentence and that exploration of non-similarity based methods may be required to further enhance the ability of computers to answer such questions.

Key words: MSR sentence completion challenge, question answering, SAT, sentence completion.

1. Introduction

Gap-filling questions are a class of questions in which a sentence is provided with one or more words replaced with a gap. The participant must then select the best word to fill each gap. In this paper, we study a specific type of gap-filling question — the sentence completion questions from the Scholastic Aptitude Test (SAT), a standardized exam used for college admissions. Each SAT sentence completion question contains a sentence containing one or two blanks, and five answer choices containing a single word (or phrase) or a pair of words (or phrases). The goal of the test-taker is to select the answer choice that best fills the gap(s) in the sentence. Two example questions are shown in Fig. 1.

The questions present a unique challenge since the answer choices are often closely related to each other, differing only in connotation. Furthermore, the questions generally test understanding of words that do not appear often in the English language, and some questions contain two blanks, in which other factors such as the relationship between the words in the blanks must also be taken into account. However, since the questions are designed to assess knowledge of English, each question contains all the information necessary to be answered without any additional context. The sentence completion questions are thus a unique class of questions through which to evaluate different language processing methods.

In addition to evaluating different language processing methods, developing methods to answer these questions have several other applications. For example, one obstacle often encountered in optical character recognition (OCR) and speech recognition is that the translation from raw input to text is imperfect. A

post-processing step is thus needed to "clean" the raw output at a lexical or semantic level. If a word is hard to recognize in the raw input, some OCR programs provide a list of possible candidates for the word [2], [3]. A gap-filling technique could be used to determine which of the candidate choices is most likely. Other possible applications of such methods also include quality control for such questions, finding search results that best fit a query, or giving word suggestions when writing documents (e.g. using mob instead of group).

The doctor does not believe in conservative approaches to teaching medicine: she uses the latest techniques, including ------ ones.
(A) outmoded (B) figurative (C) experimental
(D) cursory (E) permanent

Cookery ------ the ------ of science, for the observations of prehistoric cooks laid the foundations of early chemistry.
(A) ignored . . precision
(B) advanced . . development
(C) retarded . . supremacy
(D) aided . . decline
(E) betrayed . . methodology

Fig. 1. Two example SAT sentence completion questions [1]. The correct answers are C (top) and B (bottom).

Currently, the state-of-the-art results for solving SAT sentence completion questions is 53% correctness using a combination of latent semantic analysis (LSA) and a Good Turing language model, and the state-of-the-art results for an individual method is 46% using LSA [4]. In this study, we tested several corpus-based methods that have not yet been evaluated on SAT sentence completion questions, including co-occurrence frequencies, pointwise mutual information (PMI), and the word2vec implementations of continuous skip-gram (CSKIP) and continuous bag-of-words (CBOW) models.

Through this study, we find our co-occurrence frequency model to be the best individual method, with 52% correctness. Furthermore, our final solver has a state-of-the-art performance of 59% by combining the co-occurrence frequency, CSKIP, and CBOW models.

2. Data

In this study we used four main sources of data. We assessed the proposed methods on 108 SAT sentence completion questions obtained from official SAT practice exams between 2003 and 2014. Statistics regarding the number of blanks and the relative difficulties of the SAT sentence completion questions used are shown in Table 1 and Table 2. We note that in the answer keys of each of their practice exams, the Collegeboard provides a difficulty measure for each question. The difficulty ranges from 1 to 5, where a larger number indicates a greater difficulty. For certain years, the difficulty was measured in terms of *E*, *M*, and *H* instead of the usual 1 to 5 scale. For those years, we mapped the difficulties of *E* to 1, *M* to 3, and *H* to 5.

Table 1. Distribution of Number of Blanks

Number of Blanks	Number of Questions
One blank	48
Two blanks	60

Table 2. Distribution of Difficulty of Questions as Determined by the Collegeboard

Difficulty	Number of Questions
One	21
Two	18
Three	30
Four	14
Five	25

To train the methods, we used the GloWbE corpus of English websites. The entire corpus contains 1.8 billion words. However, since the SATs are mainly used for college admissions in the United States, we only used the subset of American websites in GloWbE, reducing the effective corpus size to approximately 300 million words.

One other dataset used to train the methods was human evaluation on how well a given answer choice fits in a sentence. To obtain the data, we used Amazon's Mechanical Turk service, an online platform for recruiting subjects to perform tasks. We randomly selected fifty questions from the 108, and for each question, five different sentences were generated by filling the blank(s) with one of the five answer choices such that each sentence corresponded to one of the answer choices. We then set up a series of tasks asking people, or "turkers," how well they thought the word(s) inserted into the blank(s) belonged in a given sentence. To ensure quality of results, we used "master turkers," or people who have consistently provided high-quality responses, and had twenty turkers provide their evaluation for each sentence. Since this information is more nuanced than just which answer choice fit best, we hoped this data would help enhance our models.

Finally, in 2011 Microsoft Research (MSR) released a set of sentence completion questions as a point of comparison for sentence completion question answering systems [5]. This data set consists of 1,040 sentence completion questions, each of which has five possible answer choices. Of the five answer choices, one is the correct answer. The sentence completion questions were constructed based on sentences selected from five of Sir Arthur Conan Doyle's *Sherlock Holmes* novels. With the sentence completion questions, MSR also provided a corpus of 19th century novels from which the methods can be trained. In addition to assessing our solvers on the SAT sentence completion questions, we assessed our final solvers on the MSR dataset as a basis of comparison.

3. Methodology

In the following section we discuss the intuition behind as well as the implementation of the five corpus-based methods explored in this study. Two of the methods, NPMI and co-occurrence frequencies, select their answer by observing the context surrounding the blank. The other three methods, LSA, CSKIP, and CBOW, select their answer by calculating the similarity between each answer choice and the words in the sentence. Before assessing the methods, we preprocessed the questions and the GloWbE corpus by removing stopwords and lemmatizing using the NLTK python library.

3.1. Pointwise Mutual Information

First introduced by Church and Hanks in 1990 [6], Pointwise Mutual Information (PMI) is a measure of association between two events. More specifically, PMI measures how much the actual co-occurrence of two outcomes *x* and *y* differs from what we would expect if we assumed that the two outcomes were independent. PMI can be expressed by the formula below:

$$PMI(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)} \tag{1}$$

As defined, PMI has an unbounded range, so we normalized the value as follows:

$$NPMI(x,y) = \left(\ln \frac{P(x,y)}{P(x)P(y)} \right) \Big/ \ln P(x,y) \tag{2}$$

Introduced by Bouma [7], this normalized PMI (NPMI) value guarantees that the values will be between –1 and 1, where NPMI equals 1 if the two terms only occur together, –1 if they only occur separately, and 0 if

their occurrences are as expected under independence.

PMI and NPMI are used often to measure the similarity and co-occurrence between two words [6]-[9]. In this study, NPMI is used to measure the co-occurrence frequency between each answer choice and the word that would be after it in the sentence. For example, in the first example in Fig. 1, the NPMI would be found between each answer choice and ones. If no word were after the blank in a sentence, then the word before the blank would be used. For each answer choice a score was thus calculated as follows:

$$\text{Score}(ans_i) = NPMI(ans_i[0], w_1) + [NPMI(ans_i[1], w_2)] \tag{3}$$

where ans_i is the ith answer choice, $ans_i[0]$ is the word to be placed in the first blank, and $ans_i[1]$ is the word to be placed in the second blank (if applicable). w_1 is the word after the first blank, and w_2 is the word after the second blank (if applicable). The probabilities were approximated based on relative word frequency in GloWbE, and Laplace smoothing was used to smooth the probabilities. The choice with the highest score was selected as the answer for each question.

3.2. Co-occurrence Frequencies

One common method used in many word similarity problems is to look at word co-occurrence counts under the assumption that the context of a word can help characterize the word itself [10]. However, the definition of what a "context" is can often vary. In this study, the "context" of a word was defined to be the five words to the right and the five words to the left of the word. Let $n(w, w_i)$ be the number of times the word w_i occurs in the context of w in the GloWbE corpus. For each word w, a function f_w was created mapping each word in the vocabulary V to its relative co-occurrence frequency with w as follows:

$$f_w(w_i) = \frac{n(w, w_i)}{\sum_{w_j \in V} n(w, w_j)} \tag{4}$$

For each answer choice ans_i, a score was calculated as follows:

$$\text{Score}(ans_i) = \sum_{w \in S} f_{ans_i[0]}(w) + [f_{ans_i[1]}(w)] \tag{5}$$

where S is the set of words in a sentence. The answer choice with the highest score was selected.

3.3. Latent Semantic Analysis

Latent Semantic Analysis (LSA) is a vector-space model that can be used to create vector representations for a given set of words. LSA has already been shown to be effective in various question-answering applications including TOEFL exam questions [11] and biology multiple-choice questions [12]. In their 2012 paper, Zweig *et al.* found that LSA was the best individual method to answer SAT sentence completion questions with an accuracy of 46% [4]. For this reason, we again consider the LSA metric here.

The key components of LSA in this application are creating and reweighting a term-document matrix and then performing singular value decomposition (SVD) to obtain a low-dimensional vector representation for each word. We created the term-document matrix D from the GloWbE corpus, and reweighted the matrix using term frequency-inverse document frequency (TF-IDF) weighting. More specifically, we used the following TF-IDF weighting scheme as defined by Platt *et al.* [13] and used by Zweig *et al.* in their implementation of LSA [4]:

$$D_{ij} = \ln(f_{ij} + 1)\ln(n/d_i) \tag{6}$$

where D_{ij} is the element in D corresponding to word i and document j, f_{ij} is the number of times word i

appears in document *j*, *n* is the total number of documents, and d_i is the number of documents that contain word *i*. After creating the reweighted matrix *D'*, we performed singular value decomposition and truncated the resulting matrices *U*, *S*, and *V* to a dimension *d* such that $U_d S_d V_d^T \approx D'$. In this case, a dimensionality of *d* = 300 was used, which was shown to work well on TOEFL exam questions [11].

The most important aspect of LSA in this application was that each row of $U_d S_d$ denotes a vector representation for a given word. We approximated the similarity between two words by taking the cosine similarity between their corresponding row vectors in $U_d S_d$. The cosine similarity metric is defined below:

$$cossim(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x} \times \boldsymbol{y}}{\|\boldsymbol{x}\| \|\boldsymbol{y}\|} \tag{7}$$

A score for each answer choice was then determined by calculating the total word similarity as defined in [4]. Letting $LSA(w_0, w_1)$ be a function returning the cosine similarity between the row vectors corresponding to w_0 and w_1 in $U_d S_d$, we used the following formula to determine the score of each answer choice:

$$\text{Score}(ans_i) = \sum_{w \in S} LSA(ans_i[0], w) + [LSA(ans_i[1], w)] \tag{8}$$

Again the answer choice with the highest score was selected.

3.4. Word2Vec

Word2vec is an implementation of two algorithms for finding vector representations of words. Created by Mikolov *et al.*, it uses neural net language models to create word vectors based on the words surrounding a given word [14], [15]. The two language models in word2vec are the Continuous Bag-of-Words Model (CBOW) and the Continuous Skip-gram Model (CSKIP). Having already shown state-of-the-art performance in many applications [14], we applied word2vec to the SAT sentence completion questions in the following manner.

3.4.1. CBOW

CBOW is a bag-of-words model trained to classify a word given the *k* words before and after a given word. Using a context window of five, we trained a CBOW model using word2vec on the GloWbE corpus. Letting $CBOW(w_0, w_1)$ be the cosine similarity between two word vectors in the CBOW model, a score for each answer choice was calculated as follows. The answer choice with the highest score was selected:

$$\text{Score}(ans_i) = \sum_{w \in S} CBOW(ans_i[0], w) + [CBOW(ans_i[1], w)] \tag{9}$$

3.4.2. CSKIP

Instead of trying to predict a word given its context, CSKIP is a model trained by predicting the *k* words before and after a given word. Again using a window of five, we trained a CSKIP model using word2vec on the GloWbE corpus. Letting $CSKIP(w_0, w_1)$ be the cosine similarity between two word vectors in the CSKIP model, a score for each answer choice was calculated as follows. The answer choice with the highest score was selected:

$$\text{Score}(ans_i) = \sum_{w \in S} CSKIP(ans_i[0], w) + [CSKIP(ans_i[1], w)] \tag{10}$$

3.5. Combination Methods

As shown in previous studies [4], [16], combining different methods can lead to remarkably better performance in answering multiple choice questions. To combine the various methods, we performed a

simple linear regression on the answer choice scores given by the various methods as well as on other heuristics. Lasso linear regression was also performed since it generally returns a sparse coefficient vector (most of the coefficients are 0). For Lasso regression, we tuned the regularizer using five-fold cross validation.

Since the AMT data provided us with human evaluation on how well an answer choice fit in a sentence, for each answer choice we regressed its AMT score to a linear combination of the method scores and other predictors. In addition to the scores predicted by NPMI, co-occurrence frequencies, LSA, CBOW, and CSKIP, the following predictors were also used.

3.5.1. 3-input LSA

Used by Zweig *et al*. in [4], the score for each answer choice in 3-input LSA was a linear combination of the following three metrics:

- Total word similarity (see (8)).
- Cosine similarity between the sum of the answer-choice vectors and sum of the word vectors in the sentence.
- Number of out-of-vocabulary terms in the answer.

3.5.2. Other heuristics

In addition to those mentioned in 3-input LSA, we also tested the following predictors:

- Sentence length.
- Number of blanks.
- Part(s) of speech before the blank(s).
- Counts of parts of speech in the question.

4. Handing Low Word Frequencies: WordNet

One problem encountered when attempting to solve the SAT sentence completion questions is that some of the words tested are used very infrequently in the English language. For example, the word *reviler* only appears twice in the GloWbE corpus. In an attempt to address this issue, we used the concept of WordNet synsets to expand the vocabulary considered.

WordNet is a lexical database that groups words into synonym sets called *synsets* [17]. Each synset contains words representing different ideas. For example, the word *board* may belong to two synsets—{*board*, *plank*} and {*board*, *committee*}—that represent two different ways *board* can be used. To alleviate the issue of low word frequencies, instead of considering the similarities between each answer choice and the sentence, we considered the similarity between the synonyms in each answer choice's synsets and the sentence. Letting syn(w) be the synonyms of w as defined above, we thus modified each of the aforementioned methods as follows.

4.1. NPMI

When calculating PMI and NPMI in (1) and (2), we replaced probabilities $P(x, y)$ and $P(x)$ with the following:

$$P^*(x,y) = \sum_{w \in \text{syn}(x)} P(w,y) \tag{11}$$

$$P^*(x) = \sum_{w \in \text{syn}(x)} P(w,y) \tag{12}$$

4.2. Co-occurrence Frequencies

Instead of using $f_w(w_i)$ in (5), $f^*_w(w_i)$ was used instead:

$$f^*_w(w_i) = \frac{\sum_{w^* \in \text{syn}(w)} n(w^*, w_i)}{\sum_{w^* \in \text{syn}(w)} \sum_{w_j \in V} n(w^*, w_j)} \tag{13}$$

4.3. LSA, CBOW, and CSKIP

In all three methods, instead of calculating the cosine similarity between an answer choice *a* and each word *w* in the sentence, we used the cosine similarity between syn(*a*) and *w*. This modification was implemented by taking the mean of the vectors corresponding to the words in syn(*a*) and finding the cosine similarity between the mean vector and the vector corresponding to *w*.

5. Experimental Results

5.1. Language Model Results

Table 3 summarizes the results obtained from using the NPMI, co-occurrence frequencies, LSA, CBOW, and CSKIP language models. In addition to analyzing the distribution of correctness of questions answered, we also measured three other statistics. For each question that was answered incorrectly, we measured the error margin and the error rank. The error margin is a measure of how close the method was to selecting the correct answer by comparing the scores of the correct answer and the answer selected by the method. The error rank similarly measures how close the method was to selecting the correct answer but instead does this by measuring the rank of the correct answer with respect to the other answers (did it have the second, third, fourth, or fifth best score). For questions that were answered correctly, we measured the correctness margin. This metric measures how by how much of a margin the method selected the correct answer by again comparing a method's scores for each answer choice. The formulas for the error and correctness margins for a question q are defined as follows:

$$\text{error margin} = \frac{q\text{score}_1 - q\text{score}[\text{selected\_}ans]}{q\text{score}_1 - q\text{score}_5} \tag{14}$$

$$\text{correctness margin} = \frac{q\text{score}_1 - q\text{score}_2}{q\text{score}_1 - q\text{score}_5} \tag{15}$$

where $q\text{score}_i$ is the *i*th highest score an answer choice received for the question, and $q$score[selected_*ans*] is the score given to the answer that the method selected for the question. Note that in the equation for correctness margin $q\text{score}_1 = q$score[selected_*ans*] since the method selected the correct answer, meaning that the correct answer choice had the highest score for the question. Table 3 displays the average error margin, error rank, and correctness margin for each method.

Table 3. Raw Results for Methods

Method	% Correct	% Incorrect by No. Blanks		% Incorrect by Difficulty					Avg. Error Margin (%)	Avg. Error Rank	Avg. Correctness Margin (%)
		1	2	1	2	3	4	5			
NPMI	**30**	77	67	76	61	83	57	72	53	3.15	28
Co-occ. Freq.	**52**	50	46	43	56	53	43	44	67	3.17	44
LSA	**39**	63	58	48	67	73	64	52	54	3.17	35
CBOW	**48**	47	58	48	33	73	50	44	55	3.09	36
CSKIP	**48**	45	60	48	44	57	57	52	45	2.93	26

Of these different methods, the co-occurrence frequencies had the best accuracy with 52% of the

questions correct while NPMI had the worst accuracy with 30% correctness. For comparison, chance performance is 20%. In these results, one interesting phenomenon to note is that the performance of these methods seem uncorrelated with the number of blanks and the difficulty of each of the questions.

5.2. WordNet Expansion Results

Table 4. Results from Using WordNet Expansion on all Answer Choices

Method	% Correct	% Incorrect by No. Blanks		% Incorrect by Difficulty					Avg. Error Margin (%)	Avg. Error Rank	Avg. Correctness Margin (%)
		1	2	1	2	3	4	5			
NPMI	**27**	67	81	81	67	80	71	64	59	3.44	30
Co-occ. Freq.	**40**	64	72	57	61	50	64	72	53	3.02	52
LSA	**36**	62	67	57	61	70	86	52	57	3.10	40
CBOW	**46**	47	63	57	50	67	57	36	43	2.91	31
CSKIP	**44**	50	63	57	61	67	50	40	45	3.12	31

Table 4 summarizes the results obtained from expanding the set of words considered using WordNet for each of the five measures. Table 5 shows the results when the WordNet expansion step was only applied to words that appeared fewer than 100 times in the corpus.

As one can see, using WordNet expansion either did not improve or actually substantially decreased the performance of all the methods. Applying WordNet expansion only to words that appeared infrequently resulted in similar performance to that without WordNet expansion as in Table 3. Due to this phenomenon, we decided not to explore using WordNet expansion in the experiments below.

Table 5. Results from Using WordNet Expansion on Low-Usage Answer Choices

Method	% Correct	% Incorrect by No. Blanks		% Incorrect by Difficulty					Avg. Error Margin (%)	Avg. Error Rank	Avg. Correctness Margin (%)
		1	2	1	2	3	4	5			
NPMI	**29**	75	67	81	67	73	50	76	53	3.15	28
Co-occ. Freq.	**52**	52	44	43	50	47	50	52	67	3.17	44
LSA	**42**	60	56	48	61	63	64	56	54	3.17	35
CBOW	**45**	50	60	52	39	60	71	52	55	3.09	36
CSKIP	**52**	45	52	38	50	57	57	40	45	2.93	26

5.3. Amazon Mechanical Turk Results

After collecting the turker responses to each of the questions, we had twenty evaluations for each answer choice as to whether the answer choice fit, could fit, or did not fit in the sentence. To quantify these values, we assigned a score of 1 to every “fit” response, 0.5 to every ``could fit'' response, and 0 to every “no fit” response. We then averaged the twenty individual evaluations to determine an average score for each answer choice. An example of the average scores for the first question in Fig. 1 is shown in Table 6.

Table 6. Human Evaluation of Answer Choices. The Correct Answer (“Experimental”) is Bolded

Answer Choice	Average Score	Standard Deviation
outmoded	0.15	0.36
figurative	0.45	0.44
experimental	**1.00**	**0.00**
cursory	0.38	0.35
permanent	0.38	0.38

For this specific question, the human turkers performed very well. All the turkers indicated that *experimental*, which was also the correct answer, fit well in the sentence. However, this data also provides valuable insight as to how humans feel how well the other answer choices belong in the sentence, thus allowing us to discern that *figurative* was the second-best option, followed by *cursory* and *permanent*, and that *outmoded* was evaluated as the worst. It makes sense that the term *outmoded* would be the least likely to fit since it is antonymic to the idea in the question that "The doctor does not believe in conservative approaches."

Overall, the human turkers performed very well on the tasks. The correct answer had the highest average score for 92% (46/50) of the questions. Furthermore, of the 46 questions, the average score given to the correct answer was 0.90, and the average difference between the scores of the correct answer and the answer with the second-highest score was 0.35.

As evidenced by the four questions for which the correct answer did not have the highest average score, the turkers were not perfect in their evaluations. However, since different people have differing opinions on how well a word fits in a sentence, and due to the strong results of the turkers on the other questions, we felt that the data collected from Amazon Mechanical Turk well-represented the approximate fit of a word in a sentence.

To assess how well each of the individual methods performed in predicting how well each answer choice fits in a sentence, we found the Pearson correlation coefficient between the score given to each answer choice and the corresponding human score. The Pearson correlation coefficient measures the linear dependence between two variables. The value of the coefficient is between 1 and -1 inclusive, where a coefficient of 1 indicates absolute positive correlation, 0 indicates no correlation, and -1 indicates absolute negative correlation. The results are given in Table 7.

Table 7. Correlation Coefficient between Method Scores and Human Scores

Method	Pearson's Correlation Coefficient
NPMI	0.15
Co-occ. Freq.	0.26
LSA	0.24
CBOW	0.35
CSKIP	0.39

The correlation coefficients and the performance of the methods seem closely tied. The CBOW, CSKIP, and the co-occurrence frequencies methods had the highest correlation coefficients and correspondingly had the three highest performances on the questions. NPMI had the lowest correlation coefficient and had the lowest performance on the questions, and similarly with the LSA method. These results further corroborate the relative quality of the different methods as well as the quality of the Amazon Mechanical Turk data.

5.4. Combination Results

Table 8. Results of Combination Methods

Method	Least Squares		Lasso	
	R^2	Test Accuracy (%)	R^2	Test Accuracy (%)
A	0.06	43	0.05	40
B	0.09	40	0.09	40
C	0.16	57	0.16	**59**
D	0.17	50	0.14	31
E	0.30	36	0.15	**59**

A: 3-input LSA; B: NPMI + Co-occ. Freq. + LSA; C: NPMI + Co-occ. Freq. + LSA + CBOW + CSKIP; D: NPMI + Co-occ. Freq. + 3-input LSA + CBOW + CSKIP; E: NPMI + Co-occ. Freq. + LSA + CBOW + CSKIP + heuristics

Using the method scores as regressors and the AMT results as the dependent variable, we ran least squares and lasso linear regressions to see if we could improve the results by combining the output of the pre-existing methods. We trained a linear model on the 50 questions used in AMT and then tested the model on the remaining 58 questions. The results of the various combination methods are shown in Table 8.

Of these various combination methods, a lasso regression of NPMI, co-occurrence frequencies, LSA, CBOW, and CSKIP performed the best, answering 59% of the questions correctly. Similarly, a lasso regression of NPMI, co-occurrence frequencies, LSA, CBOW, CSKIP, as well as the various heuristics described in the methodology section also answered 59% of the questions correctly. This value is greater than the current reported state-of-the-art performance of 53% [4].

5.5. MSR Results

Table 9. Results of Running Methods on the MSR Sentence Completion Questions

Method	SAT % correct	MSR % correct
NPMI	30	34
Co-occ. Freq.	52	38
LSA	39	28
CBOW	48	47
CSKIP	48	49
NPMI + Co-occ. Freq. + LSA + CBOW + CSKIP	59	48

The results of running our various methods on the MSR dataset are shown in Table 9. All the methods were trained on the corpus of 19th century novels provided by Microsoft while the combination method was the same model presented in the previous section due to the lack of human evaluation data on the MSR dataset.

On the MSR dataset, the NPMI, CBOW, and CSKIP methods performed similarly to how they performed on the SAT dataset whereas the other methods had a decrease in performance of about 10%.

6. Discussion

First, we note that the SAT sentence completion questions are designed to be answerable without any external context. All the semantic information required to answer a question is embedded in the question itself. To see this, let us consider the first question in Fig. 1, *The doctor does not believe in conservative approaches to teaching medicine: she uses the latest techniques, including ------- ones*. The phrases *does not believe in conservative approaches* and *uses the latest techniques* indicate that the word in the blank should be a word that has a similar meaning to *latest* but perhaps with the connotation of being even newer and cutting-edge.

This characteristic of the SAT questions most likely explains the poor performance of the NPMI method. The NPMI method only looks at a word adjacent to the blank. However, the context indicating which word belongs in the blank is embedded into the entire sentence, often several words away from the blank itself. Thus the NPMI method most likely does not have enough context to determine the word that best fits. In contrast, the co-occurrence frequencies method focuses on a wider context surrounding each blank. This perhaps also explains why it has the best performance among the individual methods.

Of the methods, CBOW and CSKIP have the strongest correlation to human evaluation on the questions. On the SAT question set, they performed nearly as well as co-occurrence frequencies, and they had the best performance on the MSR question set. Furthermore, they are fairly robust methods as well — their performance on the MSR question set is similar to that on the SAT question set. This suggests that CBOW and CSKIP are strong candidates for future studies when addressing similar questions.

In the lasso regression combining NPMI, co-occurrence frequencies, LSA, CBOW, and CSKIP, it is of note

that only the co-occurrence frequencies, CBOW, and CSKIP scores had nonzero coefficients. This indicates that among the five scores, these three scores are strong predictors for SAT sentence completion questions. Furthermore, it is interesting to note that the co-occurrence frequencies method examines the context surrounding the blank while CBOW and CSKIP analyzes the similarity between the words in the question and each answer choice. The enhanced outcome from the combination of the three scores imply that context and word-similarity methods can be effectively combined to enhance results.

In the MSR results, it is of note that the co-occurrence frequency, LSA, and the combination method performed about 10% worse on the MSR sentence completion questions than on the SAT sentence completion questions. For LSA, one possible explanation for its diminished performance is the sensitivity of LSA to various parameters [17]. Since we did not tune the parameters of LSA in this study, it is possible that the differing nature of the MSR questions resulted in a diminished performance. Next, we note that the contexts of the MSR questions do not necessarily provide enough information to answer the question. Take, for example, the following question:

Presently he emerged, looking even more than before.

a) instructive b) reassuring c) unprofitable d) flurried e) numerous

In this question, it is not clear without looking at the answer choices what the meaning of the missing word should be. Even with the answer choices, the correct answer could still be debated. For the co-occurrence frequency method, this lack of context is one possible explanation for its diminished performance. For the combination method, its drop in performance is most likely due to the fact that the combination method was trained on the SAT sentence completion questions.

7. Related Work

Currently, the state-of-the-art performance in answering SAT sentence completion questions is from the 2012 paper by Zweig *et al*. [4]. This paper explores various local and global information methods through which to answer the questions, including n-gram models, a recurring neural net model, and LSA models. They found an optimal correctness of 53% by using a linear combination of the outputs of their Good-Turing smoothed n-gram and LSA total similarity models. Beyond this study, there has been no significant work done specifically on SAT sentence completion questions. However, there has been extensive work performed on TOEFL synonym and SAT analogy questions. These studies looked at many of the methods considered here such as NPMI, LSA, and various co-occurrence measures [8]–[11]. Studies have also explored the different ways of combining the outputs of various methods [16], although in this study we only considered the baseline linear combination.

Since 2011, there have also been several studies related to the MSR dataset. Of these, perhaps the most notable is a recent 2014 study showing a state-of-the-art performance of 87.4% [18]. They achieved this performance by considering various n-gram smoothing techniques in conjunction with using the Google Web1T N-gram Count corpus, a database of approximately 1 trillion different English 5-grams based on text from the web.

8. Conclusion and Future Work

Through this study, we assessed the ability of various context-based and similarity-based methods to answer SAT sentence completion questions. We found that individually, the context-based co-occurrence frequencies method performed the best with 52% correctness, and that combining co-occurrence frequencies with the similarity-based CBOW and CSKIP methods resulted in a state-of-the-art 59% correctness. As a basis of comparison, we ran our methods on the MSR sentence completion questions. Many of the methods performed similarly, although the co-occurrence frequencies, LSA, and combination method

did not, most likely due to the differences between the SAT and MSR sentence completion questions.

From this study, we feel there are various avenues of further exploration. First, we note that each solver generally performs well on different types of questions. As shown in Fig. 2, most questions had between one and three solvers answer them correctly. There were only eight questions that all the methods answered correctly and sixteen that none answered correctly. The diversity of questions answered correctly seems to indicate that certain methods are better at solving certain questions than others. If the characteristics of such questions can be identified for each method, these characteristics can be leveraged to potentially create a conglomerate method with even better performance.

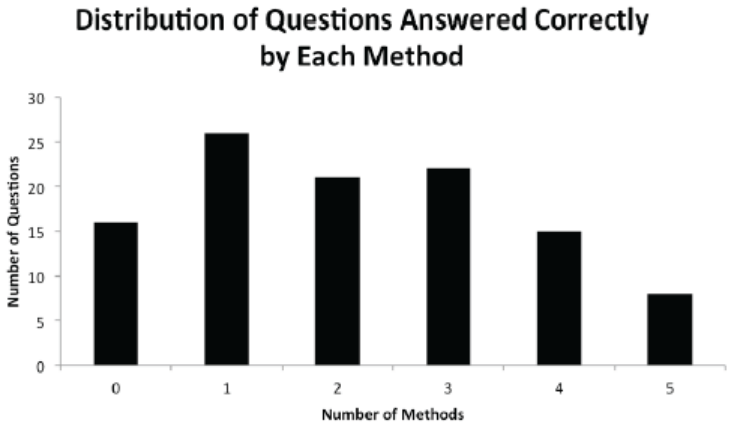

Fig. 2. Distribution of SAT questions answered correctly.

The fact that none of the methods was able to answer 16 (15%) of the questions correctly also seems to indicate a limitation of context- and similarity-based methods. Let us consider one such question:

The play closed after only a week because critics gave the performance ------- reviews.

(A) innocuous (B) caustic (C) rave (D) gaudy (E) contrite

The correct answer to this question is caustic. However, all the methods selected its antonym, rave, as their top answer. Antonyms often appear in similar contexts [19], [20]. As a result, our context-based and similarity-based methods have a hard time distinguishing between antonyms even though the semantic meanings of antonyms are completely opposite each other. Thus it could be beneficial to study other methodologies through which to determine how well a word fits in a sentence, such as by observing the syntax tree of the sentence.

Another way we hope to further explore the methods studied here is parameter tuning. Currently, we fixed many of the parameters in our methods, such as the context window in co-occurrence frequencies and the number of dimensions in LSA. However, it is possible that these parameters can significantly change the performance of the methods. For example, in [11], Landaeur and Dumais showed that there is a small range of dimensionality values for which LSA would have a sharp peak of performance on TOEFL questions. We feel that tuning these parameters could have a noticeable effect in the performance of the methods, but that the parameters used here are at least a representative sample of the relative performances of the different methods.

Acknowledgment

I would like to thank the Princeton School of Engineering and Applied Sciences without whose generous funding this project would not be possible.

References

[1] Educational-Testing-Service. (2003). Sat preparation booklet. Retrieved March 3, 2015, from

http://www.collegeboard. com/prod downloads/sat/satguide/SAT Full.pdf

[2] Jobbins, A., Raza, G., Evett, L., & Sherkat, N. (1996). Postprocessing for ocr: Correcting errors using semantic relations. *Proceedings of LEDAR. AISB 1996 Workshop on Language Engineering for Document Analysis and Recognition.* Sussex, England.

[3] Wick, M. L., Ross, M. G., & Learned-Miller, E. G. (2007). Context-sensitive error correction: Using topic models to improve ocr. *Proceedings of IEEE Ninth International Conference on Document Analysis and Recognition* (pp. 1168–1172).

[4] Zweig, G., Platt, J. C., Meek, C., Burges, C. J., Yessenalina, A., & Liu, Q. (2012). Computational approaches to sentence completion. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers: Vol. 1* (pp. 601–610). Association for Computational Linguistics.

[5] Zweig, G., & Burges, C. J. (2011). *The Microsoft Research Sentence Completion Challenge* (Technical Report MSR-TR-2011-129). Microsoft, Tech. Rep.

[6] Church, K. W., & Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational linguistics*, *16(1)*, 22–29.

[7] Bouma, G. (2009). Normalized (pointwise) mutual information in collocation extraction. *Proceedings of GSCL* (pp. 31–40).

[8] Terra, E., & Clarke, C. L. (2003). Frequency estimates for statistical word similarity measures. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: Vol. 1* (pp. 165–172). Association for Computational Linguistics.

[9] Turney, P. (2001). Mining the web for synonyms: Pmi-ir versus lsa on toefl. *Proceedings of the Twelfth Eurpoean Conference on Machine Learning* (pp. 491–502).

[10] Bullinaria, J. A., & Levy, J. P. (2007). Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, *39(3)*, 510-526.

[11] Landauer, T. K., & Dumais, S. T. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, *104(2)*, 211.

[12] Lifchitz, A., Jhean-Larose, S., & Denhie`re, G. (2009). Effect of tuned parameters on an lsa multiple choice questions answering model. *Behavior Research Methods*, *41(4)*, 1201-1209.

[13] Platt, J. C., Toutanova, K., & Yih, W.-t. (2010). Translingual document representations from discriminative projections. *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing* (pp. 251–261). Association for Computational Linguistics.

[14] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

[15] Yuan, Y., He, L., Peng, L., & Huang, Z. (2014). A new study based on word2vec and cluster for document categorization. *Journal of Computational Information Systems*, *10(21)*, 9301-9308.

[16] Turney, P., Littman, M. L., Bigham, J., & Shnayder, V. (2003). Combining independent modules to solve multiple-choice synonym and analogy problems. *Proceedings of the International Conference on Recent Advances in Natural Language Processing* (pp. 482–489).

[17] Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D., & Miller, K. J. (1990). Introduction to wordnet: An on-line lexical database*. *International Journal of Lexicography*, *3(4)*, 235–244.

[18] Lee, K., Lee, G. G., *et al.* (2014). Sentence completion task using web-scale data. *Proceedings of IEEE 2014 International Conference on Big Data and Smart Computing (BIGCOMP)* (pp. 173–176).

[19] Tanaka, T. (2002). Measuring the similarity between compound nouns in different languages using non-parallel corpora. *Proceedings of the 19th International Conference on Computational Linguistics: Vol. 1* (pp. 1–7). Association for Computational Linguistics.

[20] Wei, X., Peng, F., Tseng, H., Lu, Y., & Dumoulin, B. (2009). Context sensitive synonym discovery for web search queries. *Proceedings of the 18th ACM Conference on Information and Knowledge Management* (pp. 1585–1588). ACM.

Eugene Tang was born in 1994 and is currently pursuing a bachelor's degree in computer science at Princeton University in Princeton, New Jersey, USA.

His main interests are in statistics, machine learning and natural language processing. He has worked as a summer intern at Sandia National Laboratories, Bloomberg LP, and Two Sigma Investments.