The Microsoft Research Sentence Completion Challenge

Geoffry Zweig and Christopher J.C. Burges

Microsoft Research Technical Report MSR-TR-2011-129
February 20$^{\text{th}}$, 2011

Abstract
Work on modeling semantics in text is progressing quickly, yet currently there are few public datasets which authors can use to measure and compare their systems. This work takes a step towards addressing this issue. We present the MSR Sentence Completion Challenge Data, which consists of 1,040 sentences, each of which has four *impostor* sentences, in which a single (fixed) word in the original sentence has been replaced by an impostor word with similar occurrence statistics. For each sentence the task is then to determine which of the five choices for that word is the correct one. This dataset was constructed from Project Gutenberg data. Seed sentences were selected from five of Sir Arthur Conan Doyle's Sherlock Holmes novels, and then imposter words were suggested with the aid of a language model trained on over 500 19th century novels. The language model was used to compute 30 alternative words for a given low frequency word in a sentence, and human judges then picked the 4 best impostor words, based on a set of provided guidelines. Although the data presented here will not be changed, this is still a work in progress, and we plan to add similar datasets based on other sources. This technical report is a living document and will be updated appropriately as new datasets are constructed and new results on existing datasets (for example, using human subjects) are reported.

1 Introduction

Interest in semantic modeling for text is growing rapidly (see for example [1, 2, 3, 4]). However, currently there are few publicly available large datasets with which researchers can compare results, and those that are available focus on isolated word pairs. For example, WordSimilarity-353 [5] consists of 353 word pairs whose degree of similarity has been determined by human judges. In [6], the authors make available a test set consisting of 950 questions in which the goal is to find the word that is most opposite in meaning to another.

Geoffrey Zweig and Christopher J.C. Burges
Microsoft Research, Redmond, WA., e-mail: {gzweig, chris.burges}@microsoft.com

As a step towards addressing this problem, we present a set of 1,040 English sentences, taken from five novels written by Sir Arthur Conan Doyle. Each sentence has associated with it four *impostor* sentences, in which a single (fixed) word in the original sentence has been replaced by an impostor word with similar occurrence statistics. For each sentence the task is then to determine which of the five choices for that word is the correct one. The task is thus similar to a language SAT test. Our dataset was constructed from 19th century novel data from Project Gutenberg. We chose to use this source because of the high quality of the English, and also to avoid any copyright issues. We chose to use a single author (Conan Doyle) for the target sentences to give a consistent style of writing. We plan to construct similar datasets in the future to help explore other axes (multiple authors, and modern English, such as is typical in Wikipedia). Our data can be found at *http://research.microsoft.com/scc/*.

2 The Question Generation Process

Question generation was done in two steps. First, a candidate sentence containing an infrequent word was selected, and alternates for that word were automatically determined by sampling with an n-gram language model. The n-gram model used the immediate history as context, thus resulting in words that make "look good" locally, but for which there is no a-priori reason to expect them to make sense globally. In the second step, we eliminated choices which are obviously incorrect because they constitute grammatical errors. Choices requiring semantic knowledge and logical inference were preferred, as described in the guidelines, which we give in section 3. Note that an important *desideratum* guiding the data generation process was requiring that a researcher who knows exactly how the data was created, including knowing which data was used to train the language model, should nevertheless not be able to use that information to solve the problem. We now describe the data that was used, and then describe the two steps in more detail.

2.1 Data Used

Seed sentences were selected from five of Conan Doyle's Sherlock Holmes novels: *The Sign of the Four (1890), The Hound of the Baskervilles (1892), The Adventures of Sherlock Holmes (1892), The Memoirs of Sherlock Holmes (1894),* and *The Valley of Fear (1915)*. Once a focus word within the sentence was selected, alternates to that word were generated using a n-gram language model. This model was trained on approximately 540 texts from the Project Gutenberg collection, consisting mainly of 19th century novels. Of these 522 had adequate headers attesting to lack of copyright, and they are now available the *Sentence Completion Challenge* website.

2.2 Automatically Generating Alternates

Alternates were generated for every sentence containing an infrequent word. A state-of-the-art class-based maximum entropy n-gram model [7] was used to generate the alternates. The following procedure was used:

1. Select a word with overall frequency less than 10^{-4}. For example, we might select "extraordinary" in "It is really the most extraordinary and inexplicable business."
2. Use the two-word history immediately preceding the selected focus word to predict alternates. We sampled 150 unique alternates at this stage, requiring that they all have frequency less than 10^{-4}. For example, "the most" predicts "handsome" and "luminous."
3. If the original (correct) sentence has a better score than any of these alternates, reject the sentence.
4. Else, score each option according to how well it and its immediate predecessor predict the next word. For example, the probability of "and" following "most handsome" might be 0.012.
5. Sort the predicted words according to this score, and retain the top 30 options.

In step 3, omitting questions for which the correct sentence is the best makes the set of options more difficult to solve with a langauge model alone. However, by allowing the correct sentence to potentially fall below the set of alternates retained, an opposite bias is created: the language model will tend to assign a lower score to the correct option than to the alternates (which were chosen by virtue of scoring well). We measured the bias by performing a test using the langauge model, and choosing the *lowest* scoring candidate as the answer. This gave an accuracy of 26% (as opposed to 31%, found by taking the highest scoring candidate). Thus although there is some remaining bias for the answer to be low scoring, it is small. When a language model other than the precise one used to generate the data is used, the score reversal test yielded 17% correct. The correct polarity gave 39%. If, however, just the single score used to do the sort in the last step is used (i.e. the probability of the immediate successor alone), then the lowest scoring alternate is correct about as much as the language model itself. Neither is anywhere close to human performance.

The overall procedure has the effect of providing options which are both well-predicted by the immediate history, and predictive of the immediate future. However, in total it uses just four consecutive words, and cannot be expected to provide globally coherent alternates.

2.3 Human Grooming

The human judges (who picked the best four choices of impostor sentences from the automatically generated list of thirty) were given the following instructions:

1. All chosen sentences should be grammatically correct. For example: *He dances while he ate his pipe* would be illegal.
2. Each correct answer should be unambiguous. In other words, the correct answer should always be a significantly better fit for that sentence than each of the four impostors; it should be possible to write down an explanation as to why the correct answer is the correct answer, that would persuade most reasonable people.
3. Sentences that might cause offense or controversy should be avoided.

4. Ideally the alternatives will require some thought in order to determine the correct answer. For example:

   - *Was she his [client | musings | discomfiture | choice | opportunity] , his friend , or his mistress?*

   would constitute a good test sentence. In order to arrive at the correct answer, the student must notice that, while *"musings"* and *"discomfiture"* are both clearly wrong, the terms *friend* and *mistress* both describe people, which therefore makes *client* a more likely choice than *choice* or *opportunity*.
5. Alternatives that require understanding properties of entities that are mentioned in the sentence are desirable. For example:

   - *All red-headed men who are above the age of [800 | seven | twenty-one | 1,200 | 60,000] years , are eligible.*

   requires that the student realize that a *man* cannot be seven years old, or 800 or more. However, such example are rare: most often, arriving at the answer will still require thought, but will not require detailed entity knowledge, such as:

   - *That is his [generous | mother's | successful | favorite | main] fault , but on the whole he's a good worker.*
6. We encourage the use of a dictionary, if necessary.
7. A given sentence should only occur once. If more than one target word has been identified for a sentence (i.e. different targets have been identified, in different positions), choose the set of sentences that generates the best challenge, according to the above guidelines.

Note that the impostors sometimes constitute a perfectly fine completion, but that in those cases, the correct completion is still clearly identifiable as the most likely completion.

3 Guidelines for Use

It is important for users of this data to realize the following: since the test data was taken from five 19th century novels, the test data itself is likely to occur in the index of most Web search engines, and in other large scale datasets that were constructed from web data (for example, the Google N-gram project). For example, entering the string *That is his fault , but on the whole he's a good worker* (one of the sentence examples given above, but with the target word removed) into the Bing search engine results in the correct (full) sentence at the top position. It is important to realize that researchers may inadvertently get better results than truly warranted because they have used data that is thus tainted by the test set. To help prevent any such criticism from being leveled at a particular publication, we recommend than in any set of published results, the exact data used for training and validation be specified.

4 Baseline Results

4.1 A Simple 4-gram model

As a sanity check we constructed a very simple N-gram model as follows: given a test sentence (with the position of the target word known), the score for that sentence was initialized to zero, and then incremented by one for each bigram match, by two for each trigram match, and by three for each 4-gram match, where a match means that the N-gram in the test sentence containing the target word occurs at least once in the background data. This simple method achieved 34% correct (compared to 20% by random choice) on the test set.

4.2 Smoothed N-gram model

As a somewhat more sophisticated baseline, we use the CMU language modeling toolkit [1] to build a 4-gram language model using Good-Turing smoothing. We kept all bigrams and trigrams occurring in the data, as well as four-grams occurring at least twice. We used a vocabulary of the 126k words that occurred five or more times, and this resulted in a total of 26M N-grams. This improved by 5% absolute on the simple baseline to achieve 39% correct.

4.3 Latent Semantic Analysis Similarity

As a final benchmark, we present scores for a novel method based on latent semantic analysis. In this approach, we treated each sentence in the training data as a "document" and performed latent semantic analysis [8] to obtain a 300 dimensional vector representation of each word in the vocabulary. Denoting two words by their vectors $\mathbf{x}, \mathbf{y}$, their similarity is defined as the cosine of the angle between them:

$$\mathtt{sim}(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\| \mathbf{x} \| \| \mathbf{y} \|}.$$

To decide which option to select, we computed the average similarity to every other word in the sentence, and then output the word with the greatest overall similarity. This results in our best baseline performance, at 49% correct.

4.4 Benchmark Summary

Table 1 summarizes our benchmark study. First, for reference, we had an unaffiliated human answer a random subset of 100 questions. Ninety-one percent were answered correctly, showing that scores

[1] *http://www.speech.cs.cmu.edu/SLM/toolkit.html*

in the range of 90% are reasonable to expect. Secondly, we tested the performance of the same model (Model M) that was used to generate the data. Because this model output alternates that it assigns high-probability, there is a bias against it, and it scored 31%. Smoothed 3 and 4-gram models built with the CMU toolkit achieved 36 to 39 percent. The simple 4-gram model described earlier did slightly worse (hampered by a lack of smoothing), and the LSA similarity model did best with 49%. As a further check on this data, we have run the same tests on 203 sentence completion questions from a practice SAT exam and achieve similar results (Princeton Review, *11 Practice Tests for the SAT & PSAT*, 2011 Edition). To train language models for the SAT question task, we used 1.2 billion words of Los Angeles Times data taken from the years 1985 through 2002.

Method	**% Correct** (N=1040)
Human	91
Generating Model	31
Smoothed 3-gram	36
Smoothed 4-gram	39
Simple 4-gram	34
Average LSA Similarity	49

Table 1 Summary of Benchmarks

These results indicate that the “Holmes” sentence completion set is indeed a challenging problem, with a level of difficulty roughly comparable to that of SAT questions. Simple models based on N-gram statistics do quite poorly, and even a relatively sophisticated semantic-coherence model struggles to beat the 50% mark.

5 Conclusions and Future Work

We plan to add a similarly sized dataset based on Wikipedia, and also to present results found by asking human judges (who have only a non-electronic dictionary at hand) to perform the test. These human tests will be done in-house, since using M-Turk raises the problem that it is not clear how to construct the correct incentive (paying by the sentence alone will give poor accuracy, while paying by the correct sentence gives an incentive to taint the results by e.g. using a search engine). The in-house testing will also enable us to provide additional statistics regarding the judges’ backgrounds, for example, their level of education, and whether or not they are native-born English speakers.

References

[1] Y. Bengio, R. Ducharme and P. Vincent. *A Neural Probabilistic Language Model.* Advances in Neural Information Processing Systems, 2001.

[2] R. Collobert and J. Weston and L. Bottou and M. Karlen and K. Kavukcuoglu and P.P. Kuksa. *Natural Language Processing (almost) from Scratch.* CoRR, http://arxiv.org/abs/1103.0398, 2011.

[3] R. Socher and J. Pennington and E.H. Huang and A.Y. Ng and C.D. Manning. *Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions.* EMNLP, 2011

[4] P. Blackburn and J. Bos. *Representation and Inference for Natural Language. A First Course in Computational Semantics* . CSLI Publications, 1999.

[5] Finkelstein, L. and Gabrilovich, Y.M. and Rivlin, E. and Solan, Z. and Wolfman, G. and Ruppin, E. *Placing search in context: The concept revisited.* ACM TOIS 20(1), 2002.

[6] Saif Mohammad, Bonnie Dorr , and Graeme Hirst. *Computing Word-Pair Antonymy* EMNLP, 2008.

[7] Stanley Chen. *Shrinking Exponential Language Models.* HLT 2009.

[8] Deerwester, S. and Dumais, S.T. and Furnas, G.W. and Landauer, T.K. and Harshman, R. *Indexing by Latent Semantic Analysis.* Journal of the American Society for Information Science, Vol. 41, 1990.

Appendix: Changelog

- Feb. 20, 2012 - Corrected the description of the sampling procedure in question generation. In Step 3, a sentence was discarded if the correct answer scored best (not worst) according to the language model prediction score.
- Feb. 20, 2012 - Replaced the list of training data with a downloadable set.