CSC2611 Lab: Word embedding and semantic change

Jungeun "June" Lim | February 3, 2020

Code: https://github.com/JuneJLim/CSC2611_lab_2020

Part 1 Synchronic word embedding

Test 1. Similar word pairs

	Pearson's *r*	***p*-value**
Humans and word2vec	0.77	5.1e-14
Humans and M1	0.025	0.84
Humans and M1_plus	0.29	0.018
Humans and M2_10	0.21	0.09
Humans and M2_100	0.33	0.0068
Humans and M2_300	0.36	0.0037

Table 1. Pearson correlation coefficients between human-judged similarities of word pairs (from Table 1 of RG65) and similarities of the same word pairs based on word vector representations.

Table 1 shows that the word2vec embeddings mimic the human judgment better than the LSA vectors. However, since the two models are based on different corpora, vocabulary size, and window (context) size, it would be unfair to compare the LSA vectors constructed according to the exercise instruction and the word2vec embeddings pretrained by Google directly. Because the word2vec embeddings used in this analysis utilized much more data than the LSA vectors as summarized in Table 2, it is unsurprising that the word2vec showed better performance.

	LSA	**Word2vec**
Corpus	500 text, each consisting of about 2,000 words	100 billion words
Number of words	5,030	3 million
Window size	Only one preceding word	Unknown, but typical window size for word2vec is 5-10

Table 2. Summary of the difference between the LSA vectors and the word2vec embeddings used in this lab.

Considering how little resources the LSA vectors used, what was actually surprising to me was the performance of the LSA vectors, especially the 100-dimensional and 300-dimensional vectors. Altszyler et al. (2017) report that LSA showed better performance in their word-pairs similarity test and semantic categorization test than word2vec (skip-gram) when trained with the same corpus containing less than 1 million words. If I experimented with the word2vec trained in the same condition as how the LSA vectors were constructed, the LSA should have performed better.

LSA and word2vec have their advantages and disadvantages, and the choice should be made (and the evaluation should be done) depending on the model's specific purpose, application, and the available corpus. I will come back to this point later with more details to consider.

Test 2. Analogies

For the analogy test, I only considered the questions in which all four words are part of the 5,030 words that were used to construct the LSA vectors. By doing so, I could test the word2vec embeddings and the LSA vectors with the same set of questions. The details of the questions used for the test are shown in Table 3.

All the tests were done with 3CosAdd method, the standard for analogy tests. When given a, a^*, and b from two pairs of words $a{:}a^* :: b{:}b^*$, it suggests the closest word vector to $b - a + a^*$ as an answer for the unknown b^*.

For the test with the word2vec embeddings, I used the function evaluate_word_analogies built into the genism library. This function excludes the three input words in the given question from the answer candidates.

Analogy category	# of questions
capital-common-countries	20
capital-world	6
currency	0
city-in-state	46
family	90
Total semantic analogy questions	162
gram1-adjective-to-adverb	380
gram2-opposite	20
gram3-comparative	240
gram4-superlative	42
gram5-present-participle	272
gram6-nationality-adjective	53
gram7-past-tense	600
gram8-plural	306
gram9-plural-verbs	132
Total syntactic analogy questions	2045
Total analogy questions	2207

Table 3. The number of analogy questions from the Google analogy test set (Mikolov et al. 2013a) that were able to be tested with both the word2vec and LSA.

Analogy test (word2vec)

Test type	Accuracy	Raw Count
Semantic (excl. 3 words in questions)	0.89	144/162
Syntactic (excl. 3 words in questions)	0.68	1388/2045

Analogy test (LSA 300-dimension)

Test type	Accuracy	Raw Count
Semantic (incl. 3 words in questions)	0.0062	1/162
Semantic (excl. 3 words in questions)	0.17	28/162
Syntactic (incl. 3 words in questions)	0.0	0/2045
Syntactic (excl. 3 words in questions)	0.057	116/2045

Table 4. The accuracy on the semantic analogy test and the syntactic analogy test with the word2vec embeddings and LSA vectors.

For the test with the LSA vectors, I tried two versions of 3CosAdd method: 1) one that considers all the words as possible answers and 2) one that excludes the input words in the given question as the function evaluate_word_analogies does. The result of the test is summarized in Table 4.

It has been reported from previous studies that, if the input words are not excluded from consideration, the accuracy drops significantly; most of the suggested answers are *b* in (*a:a* :: b:b**), and *a** is also suggested as an answer quite often (Linzen 2016; Rogers, Drozd, and Li 2017). These reports are based on word2vec embeddings, but we can see from Table 4 that LSA shows the same tendency.

However, after excluding the input words, the LSA vectors still performed worse than the word2vec embeddings. I may blame the disadvantage of the limited resource again, given that previous work reports a competitive performance of LSA vectors (Gladkova, Drozd, and Matsuoka 2016; Levy, Goldberg, and Dagan 2015).

Another aspect to consider is how 3CosAdd method works and what it actually captures. Rogers et al. (2017) pointed out that the relationship among the words in an analogy question does not always show the pattern of $b - a + a^* \sim= b^*$. This study also showed that the accuracy of 3CosAdd and its modifications depend on the proximity of the answer to the inputs. A particular model and a method may perform well with some types of analogy questions among many different types of them, and it requires a closer inspection to decide whether the test set is fair and balanced.

General thoughts on improving current vector-based models in capturing word similarities

The easiest way to improve vector-based models would be increasing the corpus size. Adjusting the window size and dimension would also help, while their benefit is not as straightforward as a large and clean corpus. For instance, according to Goldberg (2017), larger windows tend to produce more topical similarities (i.e., “dog,” “bark,” and “leash” will be grouped together) while smaller windows tend to produce more functional and syntactic similarities (i.e., Poodle, Pitbull, and “Rottweiler”).

The context in LSA is based on the co-occurrence of words in a whole corpus, while the context in word2vec means only “local” contexts for each occurrence of words. GloVe, an extension of word2vec that utilizes a word co-occurrence matrix like LSA may work better for some tasks (Pennington, Socher, and Manning 2014).

As mentioned above, there is no one vector space model that works best for all purposes. Before talking about how to improve something, we would have to define what improvement means for our specific purpose and available resources to us.

Part 2 Diachronic word embedding

2-1. Measuring the degree of semantic change with three different methods

The first method I tried (Method 1) focuses only on the word of interest. It computes cosine similarity between a word’s vector from the first decade (1900) and the last decade (1990). It assumes that the less the cosine similarity is, the more the semantic change has occurred. It is the most widely used measurement in previous works on measuring the degree of semantic change.

The second method I tried (Method 2) focuses on the relationship between the word of interest and all the other words in the same time slice. It first computes cosine similarities between the word of interest and all the other words in the first decade (1900), which yields a vector of size 2000. The same is done with the last decade (1990). It assumes that the less the cosine similarity between the two vectors is, the more the semantic change has occurred.

The third method I tried (Method 3) takes a similar approach to Method 2, but it only focuses on the K nearest neighbors of the word of interest in the last decade (1990). (K should be predetermined; I tested with K=20.) It assumes that the nearest neighbors represent the meaning of the word of interest.

Method 3 first takes the K nearest neighbors (by cosine similarity) to the word of interest in the last decade (denoted by W), and go to the first decade to see how close (again, by cosine similarity) the words in W were at that time to the word of interest. (Imagine that you are traveling with a time machine!) If the meaning of the word has changed much during the 100 years, the words in W in the first decade should not be as close to the word of interest as they will be 100 years later.

Method 3 can be implemented with the neighbors of the word of interest in the first decade as well. I did not include this in the report for simplicity, but this version is also implemented in the code.

Method	Top 20 most changing words
1	programs objectives computer radio sector goals approach van shri media impact perspective patterns berkeley shift film assessment stanford challenge therapy
2	programs objectives radio approach goals computer signal film impact perspective patterns shift media challenge sector model pattern framework project gap
3	objectives radio film signal release programs approach computer media assessment model count focus intelligence intervention impact post memory framework resolution

Method	Top 20 least changing words
1	april june november february years october increase january century months daughter december god september feet week evening door payment miles
2	coast november april increase north february surface december quantity nature september east miles january june father evening consideration island explanation
3	december months days sea june afternoon april coast february father september january july wife nose autumn iowa gentleman dinner ohio

Table 5. Top 20 most changing words (above) and top 20 least changing words (below) based on three different methods proposed. Red words are the words that do not overlap with the results from the other methods (i.e., only one method identified them as top words).

	Method 1	Method 2	Method 3
Method 1	1.00		
Method 2	0.78	1.00	
Method 3	0.6	0.78	1.00

Table 6. The Pearson correlations among the three methods that measure the semantic change.

2-2. Evaluating the Accuracy of the Methods

The lack of "gold standard" makes it challenging to quantitatively evaluate methods that measure semantic changes. Few previous studies on semantic changes made their own "gold standard" by hiring human evaluators, but the dataset is small; Gulordava and Baroni (2011) let 5 human evaluators rate 100 words on a 4-point scale (0: no change; 1: almost no change; 2: somewhat change; 3: changed significantly) and Kulkarni et al. (2015) provided 3 human evaluators with 20 words and let them judge whether those words have experiences changes or not (yes or no). Both studies did not make their human evaluation data available to public.

I doubt that if a human evaluator can easily rate the degree of change that a word has experienced. Making such a decision is not as obvious as deciding if "smile" and "grin" is similar or filling out the blank in an SAT analogy question. A few rare cases like "gay" or "mouse" may not take much time for a layperson to decide whether it has experienced any change, but answering for such cases still requires some knowledge in etymology and history in general. Asking for rating the degree of change beyond yes or no would be highly subjective. Moreover, there are different types of semantic changes, and one method may be more sensitive than the other on a particular type (Hamilton, Leskovec, and Jurafsky 2016).

I tried to make my own evaluation set of words (some of them are shown in Table 7) but decided that it may not be very meaningful because of the reasons described above. When I tested with several words that experienced known changes, all the methods ranked the words quite high, at least in the upper half in terms of ranking, with varying degrees (Table 7); but I was not sure how to proceed from here and would it be ever justifiable to do so, especially without an access to the corpus that the model was trained on.

	M1	M2	M3
diet (the kinds of food - a course of food for losing weight)	682	324	776
sex (biological sex -sexual intercourse)	774	323	429
file (a folder - a collection of data stored in a computer)	27	50	74
cell (brain or prison cells - cell phone)	506	820	382
address (home address - email address)	79	63	54

Table 7. Words known for their semantic changes and their ranks in terms of the degree of semantic changes according to the three methods.

Next, I tried examining a) the words that a pair of methods most disagreed with and b) the top changing words that only one method has suggested (as shown in red in Table 5) but decided that whatever standard I choose, I would not be able to justify it as a valid way to quantitatively evaluate the methods I implemented unless I have a valid reference to test against. I considered referring to an English dictionary to count the number of definitions marked as "older use" or "archaic", but since I am observing only 100 years while dictionary definitions cover the entire history of a word, I could not justify using a dictionary as a reference.

Since I could not find a fair and reliable way to quantitatively evaluate the three methods, I did a qualitative analysis of the top 20 most changing words generated by each method. I decided to use Method 3 for the last step of the lab. Compared to other methods, its top 20 most changing words contain more verbs (i.e., fewer words that are only used as nouns); release, count, focus, post are words that only appear in the result of Method 3. In addition, it does not contain proper nouns as in the result of Method 1. I liked Method 3's balance and relative easiness of possible interpretations. (e.g., "resolution" and "memory" may have exper-

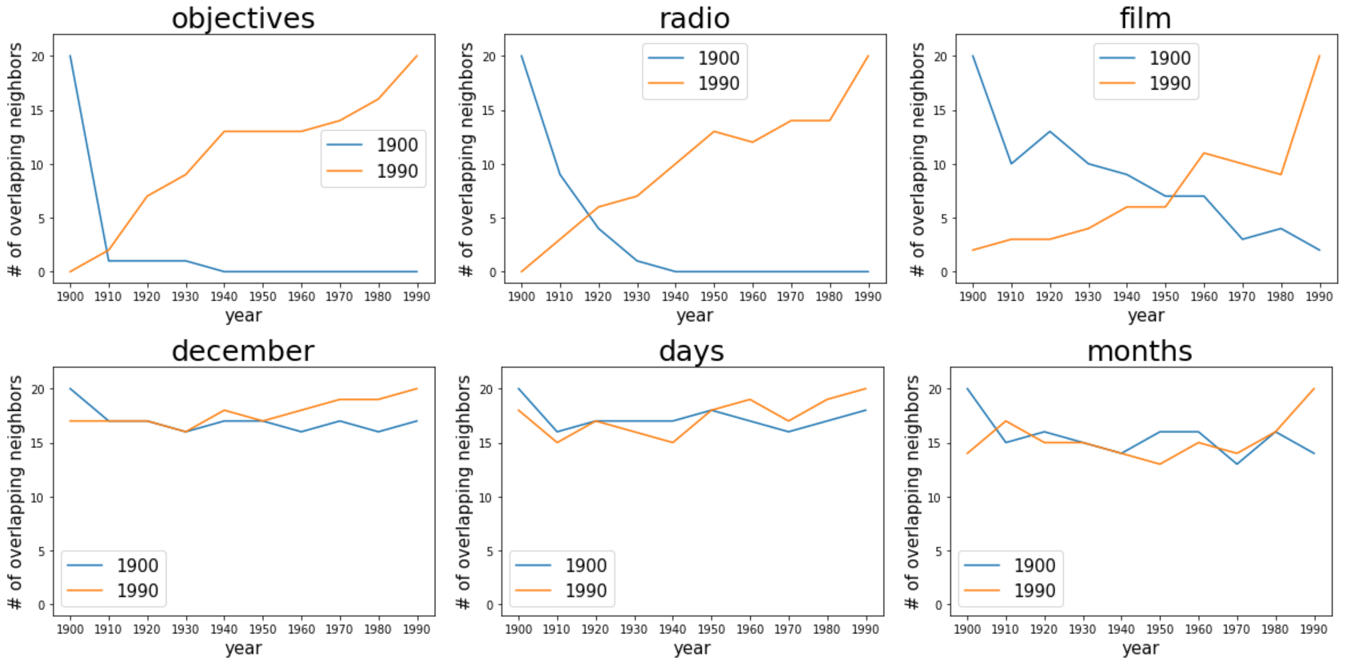

Figure 1. The blue line shows how the number of overlapping words in 20 nearest neighbors of a word in 1900 has changed across 100 years. The yellow line shows how the number of overlapping words in 20 nearest neighbors of a word in 1900 has changed across 100 years. The three charts above are the top three most changing words extracted from Method 3. The three charts below are the top three least changing words extracted from Method 3.

-ienced changes because of the development of computers; on the other hand, it was difficult for me to imagine the semantic changes that "Stanford" and "Berkeley", two of the top 20 changing word according to Method 1, have experienced without looking at the corpus.)

2-3. Detecting the Point of Semantic Change

Figure 1 shows a graph for a word. A line connects 10 points, each for a decade. The points connected with blue lines denote the number of overlapping words in 20 nearest neighbors of a word in 1900 (the first decade). The points connected with yellow lines denote the number of overlapping words in 20 nearest neighbors of a word in 1990 (the last decade). In addition to the top three most changing words, I also draw graphs for the top three least changing words for comparison.

The point where the blue line and the orange line crosses may be interpreted as a moment when the word began to be used more frequently in the current meaning and less frequently in the past meaning. For instance, we may suspect that "objectives" began to be used more frequently in a new sense around 1910, "radio" around 1920, and "film" around 1950. The crossing points are consistent between 20-150 neighbors and do not change much outside of this range with these examples.

Alternatively, the time periods in which a steep slope is observed may be considered as the points of semantic change. For instance, these graphs may imply that "objectives" has undergone an abrupt change until 1940 and "film" has experienced a drastic change after 1980.

One limitation of this method is that it assumes a one-way change from A to B, or in other words, the meaning in 1900 to the meaning in 1990. More complex changes such as A to B to C cannot be captured with this approach, even though such a case would be rare.

References

Altszyler, Edgar, Mariano Sigman, Sidarta Ribeiro, and Diego Fernández Slezak. 2017. "Comparative Study of LSA vs Word2vec Embeddings in Small Corpora: A Case Study in Dreams Database." *Consciousness and Cognition* 56 (November): 178–87. https://doi.org/10.1016/j.concog.2017.09.004.

Gladkova, Anna, Aleksandr Drozd, and Satoshi Matsuoka. 2016. "Analogy-Based Detection of Morphological and Semantic Relations with Word Embeddings: What Works and What Doesn't." In *Proceedings of the NAACL Student Research Workshop*, 8–15. San Diego, California: Association for Computational Linguistics. https://doi.org/10.18653/v1/N16-2002.

Goldberg, Yoav. *Neural Network Methods in Natural Language Processing.* San Rafael: Morgan & Claypool Publishers, 2017.

Gulordava, Kristina, and Marco Baroni. 2011. "A Distributional Similarity Approach to the Detection of Semantic Change in the Google Books Ngram Corpus." In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, 67–71. Edinburgh, UK: Association for Computational Linguistics. https://www.aclweb.org/anthology/W11-2508.

Hamilton, William L., Jure Leskovec, and Dan Jurafsky. 2016. "Cultural Shift or Linguistic Drift? Comparing Two Computational Measures of Semantic Change." In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2116–2121. Austin, Texas: Association for Computational Linguistics. https://doi.org/10.18653/v1/D16-1229.

Kulkarni, Vivek, Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2015. "Statistically Significant Detection of Linguistic Change." In *Proceedings of the 24th International Conference on World Wide Web*, 625–635. WWW '15. Florence, Italy: International World Wide Web Conferences Steering Committee. https://doi.org/10.1145/2736277.2741627.

Levy, Omer, Yoav Goldberg, and Ido Dagan. 2015. "Improving Distributional Similarity with Lessons Learned from Word Embeddings." *Transactions of the Association for Computational Linguistics* 3: 211–225. https://doi.org/10.1162/tacl_a_00134.

Linzen, Tal. 2016. "Issues in Evaluating Semantic Spaces Using Word Analogies." In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, 13–18. Berlin, Germany: Association for Computational Linguistics. https://doi.org/10.18653/v1/W16-2503.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. "Efficient Estimation of Word Representations in Vector Space." *ArXiv:1301.3781 [Cs]*, September. http://arxiv.org/abs/1301.3781.

Mikolov, Tomas, Wen-tau Yih, and Geoffrey Zweig. 2013b. "Linguistic Regularities in Continuous Space Word Representations." In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 746–751. Atlanta, Georgia: Association for Computational Linguistics. https://www.aclweb.org/anthology/N13-1090.

Pennington, Jeffrey, Richard Socher, and Christopher Manning. 2014. "Glove: Global Vectors for Word Representation." In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Doha, Qatar: Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1162.

Rogers, Anna, Aleksandr Drozd, and Bofang Li. 2017. "The (Too Many) Problems of Analogical Reasoning with Word Vectors." In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, 135–148. Vancouver, Canada: Association for Computational Linguistics. https://doi.org/10.18653/v1/S17-1017.