Causally Denoise Word Embeddings Using Half-Sibling Regression

Zekun Yang[*], **Tianlin Liu**[♯]

[*]Department of Information Systems, College of Business
City University of Hong Kong, Hong Kong SAR, China.
[♯]Friedrich Miescher Institute for Biomedical Research
Maulbeerstrasse 66, 4058 Basel, Switzerland.
zekunyang3-c@my.cityu.edu.hk, tianlin.liu@fmi.ch

Abstract

We introduce a novel word vector postprocessing scheme under a *causal inference* framework. Concretely, the postprocessing pipeline is realized by Half-Sibling Regression (HSR), which allows us to identify and remove confounding noise contained in word vectors. Compared to previous work, our proposed method has the advantages of interpretability and transparency due to its causal inference grounding. Evaluated on a battery of standard lexical-level evaluation tasks and downstream sentiment analysis tasks, our method reaches state-of-the-art performance.

Word Vector Postprocessing

Word vector postprocessing methods enhances the general quality of word vectors. Modern word vector postprocessing methods can be broadly divided into two streams: (1) **lexical** and (2) **spatial** approaches.

The Lexical Approach: The lexical approach uses lexical relational resources to enhance the quality of word vectors. These lexical relational resources specify semantic relationships of words such as synonym and antonym relationships. A shortcoming of the lexical approach is that it is unable to postprocess out-of-database word vectors.

The Spatial Approach: The general principle of the spatial approach is to enforce word vectors to be more "isotropic", i.e., more spread out in space. This goal is usually achieved by flattening the spectrum of word vectors. One major downside of the spatial approach is its lack of direct interpretability [1].

Our improvement over previous methods: We go beyond the lexical and spatial schemes and introduce a novel *causal inference approach* for postprocessing word vectors. This approach identifies and removes confounding noise of word vectors using Half-Sibling Regression (HSR) method [2].

Half-Sibling Regression

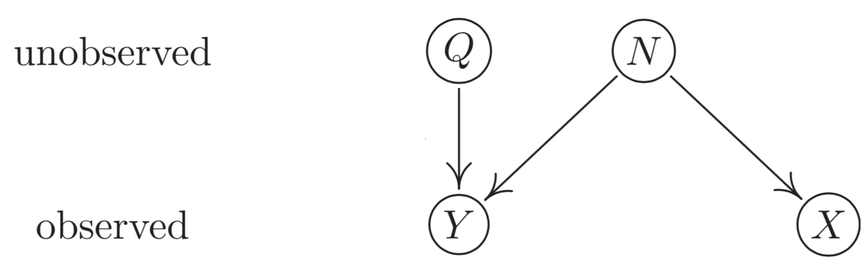


Figure 1: *The causal graph for HSR (adapted from [2]).*

Assume Y is a noisy variable influenced by the unobservable noise source N, and we wish to retrieve the "clean" variable Q from Y. Furthermore, we are given the observable variable X, which is independent of Q but also affected by the noise N.

How to reconstruct the quantities taken by Q by leveraging the underlying statistical dependency structure in Figure 1? HSR provides a simple yet effective solution to this question – It estimates Q via its approximation $\hat{Q}$, which is defined as

$$\hat{Q} := Y - \mathbb{E}[Y \mid X]. \tag{1}$$

Since X is independent of Q, X is *not* predictive to Q or Q's influence on Y. However, X is predictive to Y, because X and Y are both influenced by the *same* noise source N. When predicting Y based on X realized by the term $\mathbb{E}[Y \mid X]$, since those signals of Y coming from Q cannot be predicted by X, only those noise contained in Y coming from N could be captured. To reconstruct Q from Y, we can therefore remove the captured noise $\mathbb{E}[Y \mid X]$ from Y.

References

[1] J. Mu and P. Viswanath. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations*, 2018.

[2] B. Schölkopf, D. W. Hogg, D. Wang, D. Foreman-Mackey, D. Janzing, C. Simon-Gabriel, and J. Peters. Modeling confounding by half-sibling regression. Proceedings of the National Academy of Sciences, 2016.

Full paper **Code**

HSR Algorithm for Word Vector Postprocessing

Input: (i) $\{v_i^Y\}_{i=1}^K$: a collection of K content-word vectors, each of dimension n; $\mathbf{V}^Y$ is a $n \times K$ matrix whose columns are from $\{v_i^Y\}_{i=1}^K$. (ii) $\{v_i^X\}_{i=1}^P$: a collection of P function-word vectors, each of dimension n; $\mathbf{V}^X$ is a $n \times P$ matrix whose columns are from $\{v_i^X\}_{i=1}^P$. (iii) Regression constants $\alpha_1, \alpha_2 > 0$.

Postprocess content-word vectors:
Step 1.1: *Identify noise contained in content-word vectors*: Estimate a weight matrix $\mathbf{W}_1$ such that

$$\mathbf{V}^Y \approx \mathbf{V}^X \mathbf{W}_1,$$

with ridge regression

$$\mathbf{W}_1 = \left((\mathbf{V}^X)^\top \mathbf{V}^X + \alpha_1 \mathbf{I}\right)^{-1} (\mathbf{V}^X)^\top \mathbf{V}^Y.$$

Step 1.2: *Remove noise contained in content-word vectors*:

$$\hat{\mathbf{V}}^Y := \mathbf{V}^Y - \mathbf{V}^X \mathbf{W}_1.$$

Postprocess stop-word vectors:
Step 2.1: *Identify noise contained in stop-word vectors*: Estimate a weight matrix $\mathbf{W}_2$ such that

$$\mathbf{V}^X \approx \mathbf{V}^Y \mathbf{W}_2,$$

with ridge regression

$$\mathbf{W}_2 = \left((\mathbf{V}^Y)^\top \mathbf{V}^Y + \alpha_2 \mathbf{I}\right)^{-1} (\mathbf{V}^Y)^\top \mathbf{V}^X.$$

Step 2.2: *Remove noise contained in stop-word vectors*:

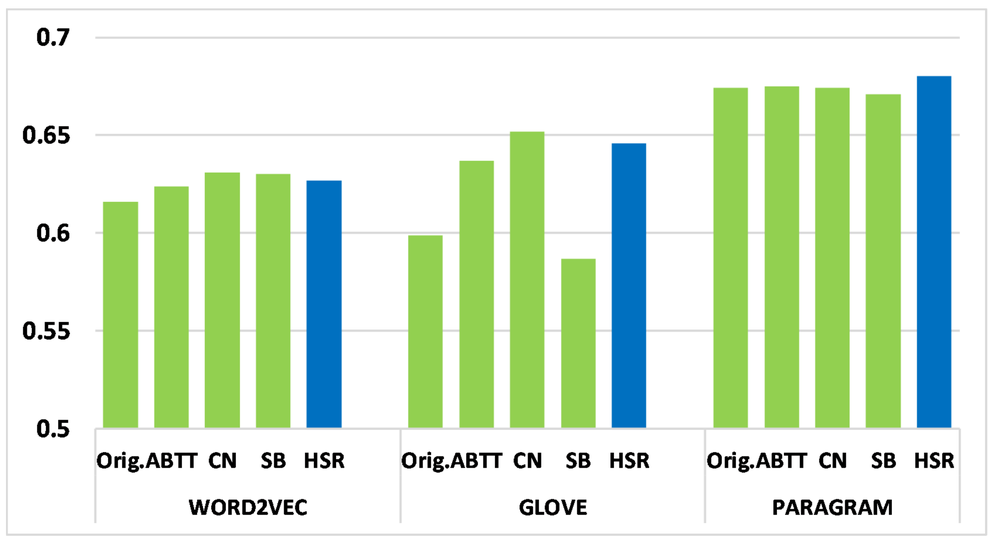
$$\hat{\mathbf{V}}^X := \mathbf{V}^X - \mathbf{V}^Y \mathbf{W}_2.$$

Output: (i) HSR postprocessed content-word vectors $\{\hat{v}_i^Y\}$, which are columns of $\hat{\mathbf{V}}^Y$; (ii) HSR postprocessed stop-word vectors $\{\hat{v}_i^X\}$, which are columns of $\hat{\mathbf{V}}^X$.

Algorithm 1: *HSR algorithm for word vector postprocessing*

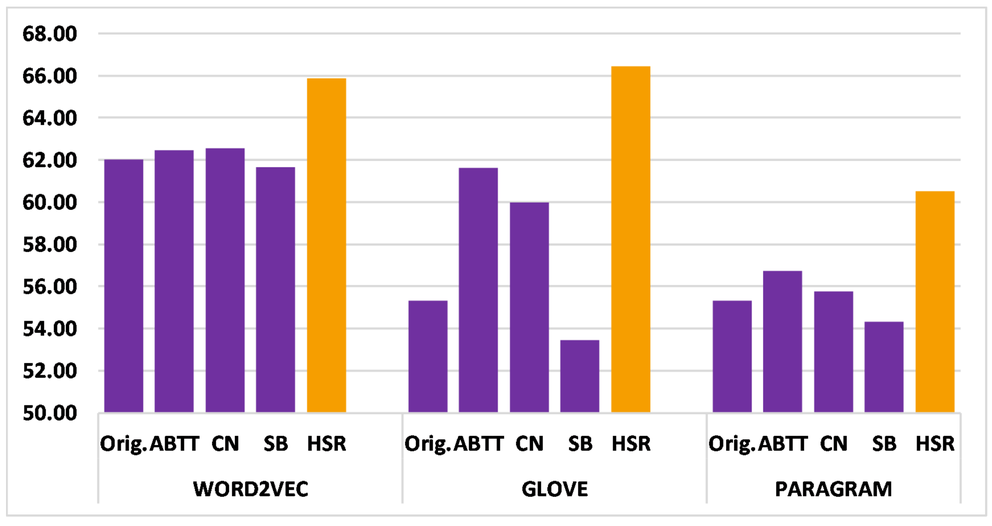
Experiments

Word Similarity



Spearman's rank correlation coefficient averaged across seven word similarity tasks.

Semantic Textual Similarity



Pearson correlation coefficient averaged across 20 semantic textual similarity tasks.

The 34th AAAI Conference on Artificial Intelligence (AAAI), New York, New York, USA, 2020