Word Embeddings with BERT

Contents

WHAT ARE WORD EMBEDDINGS? 1

DIFFERENCES BETWEEN WORD EMBEDDINGS USING BERT AND USING WORD2VEC 1

WORD2VEC 1
BERT 1

RESULTS: HTML FILES 2

HTML VISUAL 2
WHAT IS T-SNE? 2
THE HTML FILES 3
ZOOMING IN INTO THE CHART AREA 3
HOW DO I GET BACK TO THE ORIGINAL CHART? 4
X-Y AXES: WHAT DO THEY MEAN? 5

REFERENCES 5

What are word embeddings?

Word embeddings are vector values corresponding to specific words. There are many different models that can construct these vector values (BERT models, Word2Vec, etc.). Words are converted into such vectors so that they can be plotted easily on XY or XYZ axes, and we can analyze words with close relations to one another in a semantic space.

Differences between Word Embeddings using BERT and using Word2Vec

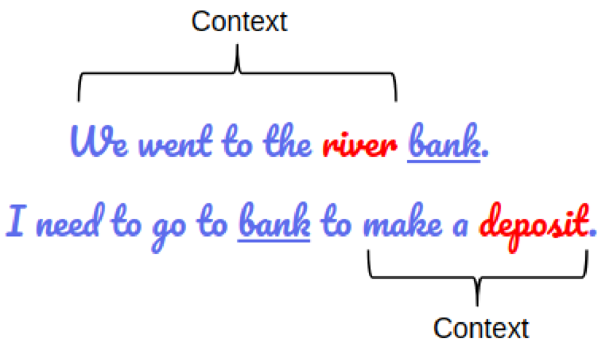
Word2Vec

The biggest difference is that Word2Vec is a *context-independent* model. This means that there is only one vector representation for a specific word. The context in which it is used is not considered. “Different senses of the word (if any) are combined into one vector” (Gupta, 2020).

BERT

Contrary to Word2Vec, BERT models for word embeddings are *context-dependent*. Thus, we are able to get multiple vector representations for the same word, based on the context the has been used in.

Take the following example from Gupta (2020):


Context
We went to the river bank.
I need to go to bank to make a deposit.
Context

“The word *bank* is being used in two different contexts — a) financial entity b) land along the river (geography). Word2Vec will generate the same single vector for the word *bank* for both the sentences. Whereas BERT will generate two different vectors for the word *bank* being used in two different contexts. One vector will be similar to words like money, cash etc. The other vector would be similar to vectors like beach, coast etc.”

Results: HTML files

Word embeddings with BERT in the NLP Suite will create two **html files**. One will show words when you hover over a plot, the other will already display words on the graph.

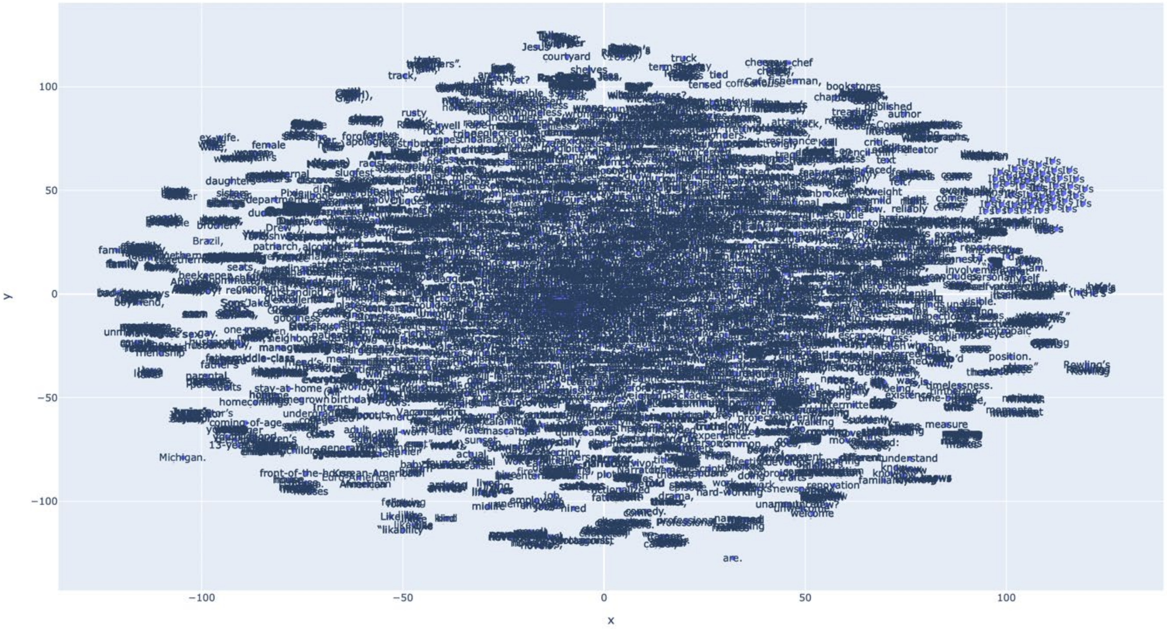
HTML visual

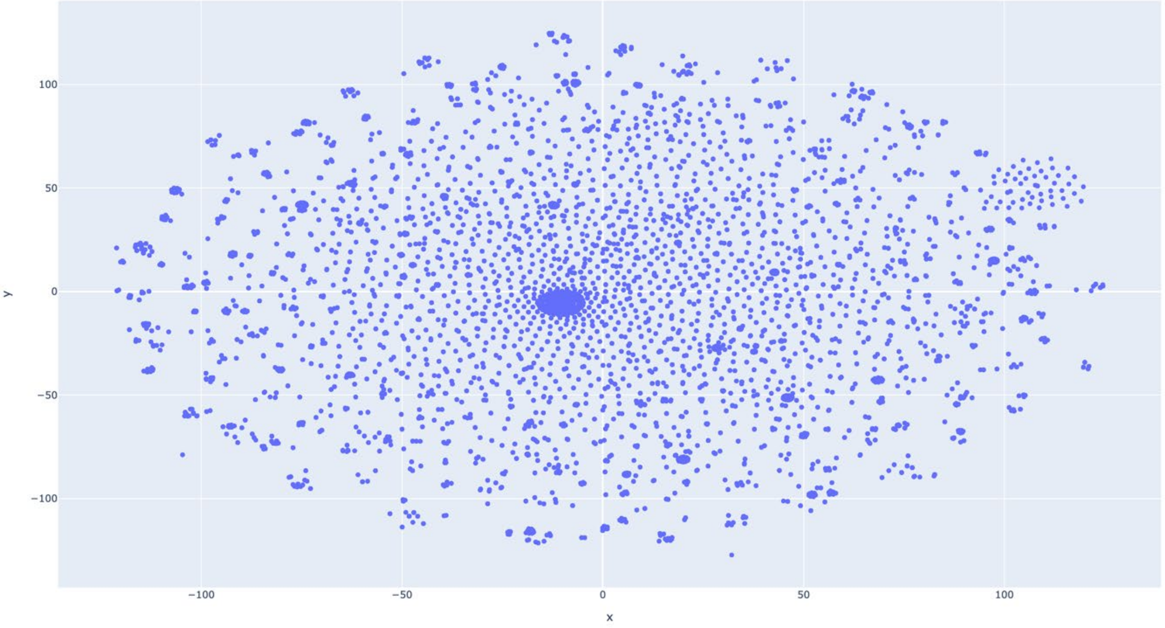
The resulting vectors for each word will be displayed in either two-dimensional or three-dimensional Cartesian space in an html file using t-SNE. When you hover over with the mouse on specific points, you can display words. Look carefully to see whether the spatial distribution of words suggests connections you had not thought about.

What is t-SNE?

T-SNE is a machine learning algorithm for visualization based on nonlinear dimensionality reduction technique. The basic idea of t-SNE is to reduce dimensional space keeping relative pairwise distance between points, so points which were initially far from each other are also located far away, and close points are also converted to close ones. However, the t-SNE has a non-convex objective function, which is minimized using a gradient descent optimization with random initiation, so different runs produce slightly different results.

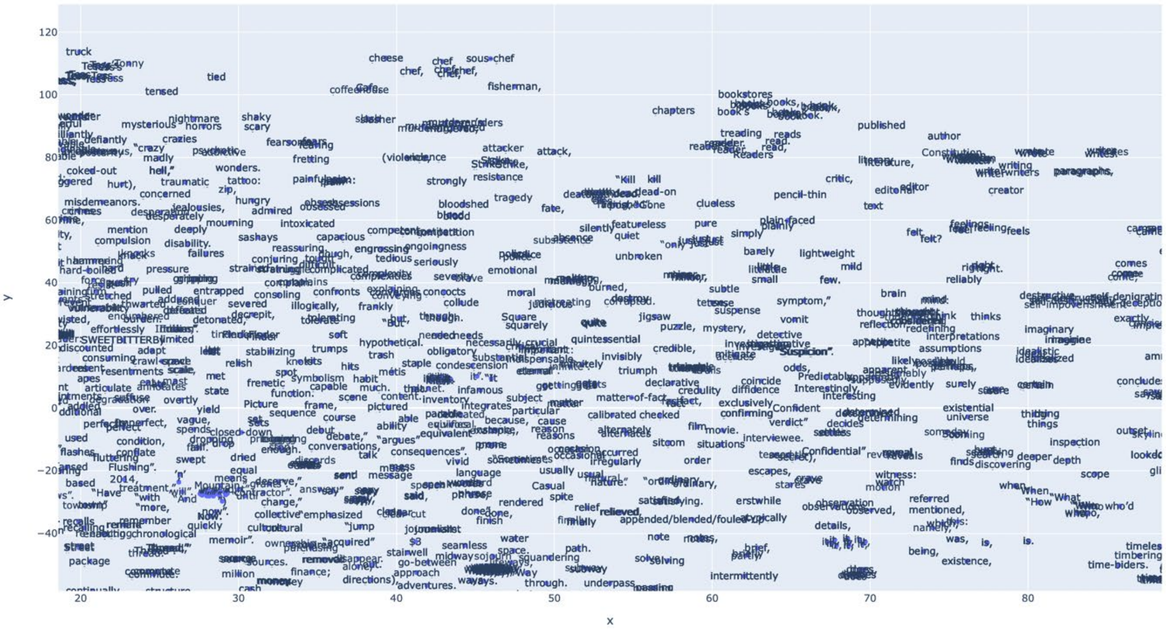
The html files



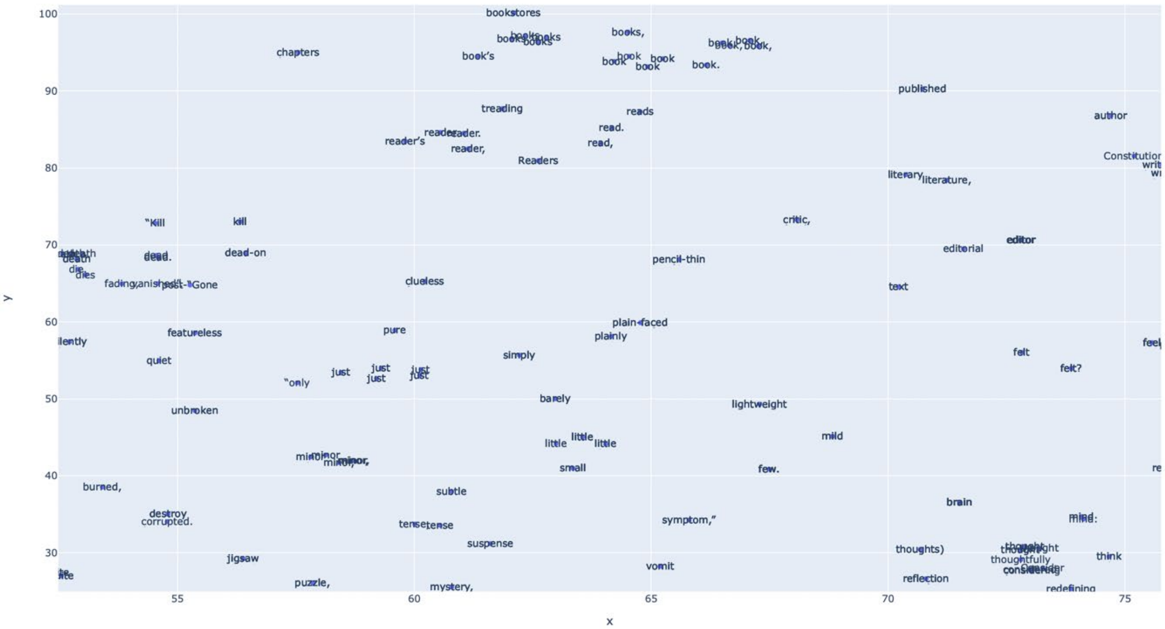


Zooming in into the chart area

Unfortunately, these visual outputs look pretty messy. **Hard to tell which words come together**. Fortunately, there is an easy solution to that problem. With your mouse, just selected an area of the chart you are interested in. It will zoom into the words in the selected area. Things start to look better (and the smaller the area you draw with your mouse and the easier it will be to read the words in the new selected chart area).



And you can repeat the same operation on the new chart to zoom in further. You can do that to your heart's fill.



How do I get back to the original chart?

Click on the Refresh button to reload the page and display the original image.

X-Y axes: What do they mean?

The distribution of words in the Cartesian plane, negative-positive axes does not mean anything in T-SNE. T-SNE only cares about the pairwise distance, so the X-Y axes of the plot do not have any specific meaning.

References

TIPS_NLP_Word2Vec.pdf

Gupta, Lavanya. 2020. “Differences Between Word2Vec and BERT.” https://medium.com/swlh/differences-between-word2vec-and-bert-c08a3326b5d1

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. “Efficient Estimation of Word Representations in Vector Space.” *arXiv*. https://arxiv.org/abs/1301.3781