Interpreting Word Embeddings with Eigenvector Analysis

Anonymous Author(s)
Affiliation
Address
e-mail

Abstract

Dense word vectors have proven their values in many downstream NLP tasks over the past few years. However, the dimensions of such embeddings are not easily interpretable. Out of the $d$-dimensions in a word vector, we would not be able to understand what high or low values mean. Previous approaches addressing this issue have mainly focused on either training sparse/non-negative constrained word embeddings, or post-processing standard pre-trained word embeddings. On the other hand, we analyze conventional word embeddings trained with Singular Value Decomposition, and reveal similar interpretability. We use a novel eigenvector analysis method inspired from Random Matrix Theory and show that semantically coherent groups not only form in the row space, but also the column space. This allows us to view individual word vector dimensions as human-interpretable semantic features.

1 Introduction

Understanding words has a fundamental impact on many natural language processing tasks, and has been modeled with the Distributional Hypothesis [1]. Dense $d$-dimensional vector representations of words created from this model are often referred to as word embeddings, and have successfully captured similarities between words, such as word2vec and GloVe [2, 3]. They have also been applied to downstream NLP tasks as word representation features, ranging from sentiment analysis to machine translation [4, 5].

Despite their widespread popularity in usage, the dimensions of these word vectors are difficult to interpret [6]. Consider $w_{\text{president}} = [0.1, 2.4, 0.3]$ as the 3-dimensional vector of “president” from word2vec. In this 3-dimensional space (or the row space), semantically similar words like “minister” and “president” are closely located. However, it is unclear what the dimensions represent, as we do not know the meaning of the 2.4 in $w_{\text{president}}$. It is difficult to answer questions like ‘what is the meaning of high and low values in the columns of $W$’ and ‘how can we interpret the dimensions of word vectors’. To address this problem, previous literature focused on the column space by either training word embeddings with sparse and non-negative constraints [7-9], or post-processing pre-trained word embeddings [6, 10, 11]. We instead investigate this problem from a random matrix perspective.

In our work, we analyze the eigenvectors of word embeddings obtained with truncated Singular Value Decomposition (SVD) [12, 13] of the Positive Pointwise Mutual Information (PPMI) matrix [14]. Moreover, we compare this analysis with the row and column space analysis of Skip Gram Negative Sampling (SGNS), a model used to train word2vec [15]. From the works of [16] proving that both SVD and SGNS factorizes and approximates the same matrix, we hypothesize that a study of the principal eigenvectors of the PPMI matrix reflects the information contained in SGNS.

Contributions: Without requiring any constraints or post-processing, we show that the dimensions of word vectors can be interpreted as semantic features. In doing so, we also introduce novel word embedding analysis methods inspired by the literature of eigenvector analysis techniques from Random Matrix Theory.

2 Related Work

Recently, there have been several works that have shown similar results in semantic grouping among the column values. Several of these algorithms proposed to train non-negative sparse interpretable word vectors [7,9,17]. Furthermore, [6] also proposed methods to post-process pre-trained word vectors with non-negativity and sparsity constraints. However, their vectors were optionally binarized, which is difficult to interpret intensity than real-values. [10] has proposed to overcome these limitations by simply training a rotation matrix to transform pre-trained word2vec and GloVe, without being sparse or binary. Finally, [11] post-trained the pre-trained word embeddings with k-sparse autoencoders with similar constraints to [6].

While these methods were able to successfully achieve interpretability in the column space evaluated with word intrusion detection tests, they either enforced sparsity and non-negativity constraints, or required extensive post-processing. Furthermore, they focused less on the analysis and discussion on the actual meanings of the columns despite their pursuit of interpretable dimensions. Hence, in our work, we put more emphasis on such implications with conventional algorithms without any extra constraints or post-processing steps.

3 Methodology

3.1 Notations

We define the Positive Pointwise Mutual Information (PPMI) matrix as \( \mathbf{M}^\text{PPMI} \), the set of unique words as vocabulary \( \mathcal{V} \), and word embedding matrices created from SVD and SGNS as \( \mathbf{W}^\text{SVD} \) and \( \mathbf{W}^\text{SGNS} \). The \( k \)-th largest eigenvalue and corresponding eigenvector of \( \mathbf{M}^\text{PPMI} \) are denoted as \( \lambda_k \) and \( \mathbf{v}_k \in \mathbb{R}^{|\mathcal{V}|} \), and the \( k \)-th column of \( \mathbf{W}^\text{SGNS} \) as \( \mathbf{v}_k \in \mathbb{R}^{|\mathcal{V}|} \). The word vectors are denoted \( \mathbf{w}^\text{word}_k \) or \( \mathbf{w}^\text{SVD}_k \), but when context is clear or does not matter, we simply use \( \mathbf{w}_\text{word} \). Note that we often use the term "eigen" when and "singular" interchangeably because \( \mathbf{M}^\text{PPMI} \) is defined as a square matrix.

3.2 Positive Pointwise Mutual Information (PPMI) Matrix

Each entry of a co-occurrence matrix \( \mathbf{M} \) represents the co-occurrence counts of words \( w_i \) and \( c_j \) in all documents in the corpus. However, raw co-occurrence counts have been known to underperform than other transformed variants [16]. Pointwise Mutual Information (PMI) [14] instead transforms matrix by measuring the log ratio between the joint probability of \( w \) and \( c \) when assuming independence of the two and not.

\[
\text{PMI}(w,c) = \log \frac{P(w,c)}{P(w)P(c)} = \log \frac{\#(w,c)}{\#(w)\#(c)}
\]

The problem of this association measure is when dealing with never observed pairs which result in \( \text{PMI}(w,c) = \log 0 \). To cope with such, Positive Pointwise Mutual Information has been used to map all negative values to 0 from the intuition that positive associations are often more informative in downstream NLP tasks [16].

\[
\text{PPMI}(w,c) = \max(\text{PMI}(w,c), 0)
\]

3.3 Truncated Singular Value Decomposition (SVD)

Truncated SVD (we will further refer this as simply SVD), which is equivalent to maximum variance Principal Component Analysis (PCA) and has been popularized by Latent Semantic Analysis (LSA) [13], factorizes the PPMI matrix as \( \mathbf{M}^\text{PPMI} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^\dagger \) and truncates to \( d \) dimensions. Following the works of [18], the word embedding matrix is taken as \( \mathbf{W} = \mathbf{U}_d \cdot \mathbf{S} \). We discuss the effect of this in Section 6.2.
3.4 Skip-Gram with Negative Sampling (SGNS)

Unlike PPMI and SVD which gives exact solutions, the word2vec Skip-Gram model, proposed by [2], trains two randomly initialized word embedding matrices $W$ and $C$ with a neural network.

\[ P(C_j|W_i) = \text{Softmax}(W_i \cdot C_j), \quad \text{where } \text{Softmax}(W_i \cdot C_j) = \frac{e^{W_i \cdot C_j}}{\sum_k e^{W_i \cdot C_k}} \]

The intuition is to basically maximize the dot product between "similar" word and context pairs, and minimize the dot product between wrong pairs. The Softmax function is simply a generalized version of the logistic function to multi-class scenario. However, the normalization constant which computes the exponentials of all context words, is very computationally expensive when the vocabulary size is large. Hence, [15] proposed Skip Gram with Negative Sampling (SGNS) to simplify the objective using negative sampling.

3.5 Eigenvector Analysis Methods

We borrow intuitions from the Random Matrix Theory literature to analyze eigenvectors of $M^{PPMI}$. We analyze the distributions of eigenvectors, calculate the Inverse Participation Ratios (IPR) to quantify the ratio of significant elements and measure structural sparsity, and qualitatively interpret the significant elements.

**Distribution of Eigenvector:** The empirical distribution of eigenvector elements $u^k_i$ is compared with a Normal distribution $N(\mu^k_i, \sigma^2) \!$ to measure normality of the eigenvectors. [19] have shown that eigenvectors deviating from Gaussian contain genuine correlation between stocks, while also revealing a global bias that represented newsbreaks influencing all stocks. We search for similar patterns in Section 5.1.

**Inverse Participation Ratio:** The Inverse Participation Ratio (IPR) of $u^k$, denoted as $I^k$, quantifies the inverse ratio of significant elements in the eigenvector $u^k$.

\[ I^k \triangleq \sum_{i=1}^{\mid V \mid} \frac{|u^k_i|^4}{|u^k_i|}, \]

where $u^k_i$ is the $i$-th element of $u^k$. The intuition of IPR can be illustrated with two extreme cases. First, if all elements of $u^k$ have same values $1/\sqrt{\mid V \mid}$, then $I^k$ is simply $1/\mid V \mid$, with reciprocal $1/I^k$ being $\mid V \mid$. This means that all $\mid V \mid$ elements contribute similarly. On the other hand, a one-hot vector with only one element as one, and the rest as zero, $u^k$ will have an IPR value of one (also same for reciprocal). Hence, the reciprocal, $1/I^k$, measures the ratio of significant participants in $u^k$. In short, the larger the $I^k$, the smaller the ratio of participation, and the sparser the vector, in turn, reflecting structural sparsity of $u^k$. Furthermore, as $1/I^k \in [1, \mid V \mid]$, dividing this reciprocal with $\mid V \mid$ will yield the sparsity of a given vector $u^k \in \mathbb{R}^{\mid V \mid}$.

**Visualization of Top Eigenvector elements:** As $u^k, v^k \in \mathbb{R}^{\mid V \mid}$, we can map each index of the vectors to a word in the vocabulary $V$. Hence, we investigate the dimensions and their indices (or words) with the largest absolute values and search for semantic coherence. Similar approaches with financial data have shown to group stocks from same industries or nearby regions [19], and with genetic data, revealed important co-evolving genes in gene co-expression networks [20].

4 Experimental Setup

4.1 Training

**English Wikipedia** We use the English Wikipedia dump cleaned by adapting Matt Mahoney’s Perl script [3] which has also been used by [2]. Removing most of the noisy non-alphanumerics, such as XML tags, the dataset size effectively reduced from approximately 66GB to 25GB, containing around 3.4B tokens. The vocabulary size is approximately 346K as we only consider words with at least 100 occurrences.

[^1]: [https://dumps.wikimedia.org/enwiki/20180420/]
[^2]: At the bottom of [http://mattmahoney.net/dc/textdata.html]
SGNS and SVD We adapt the code from the hyperwords released by [18] to train both $W_{SVD}$ and $W_{SGNS}$. For $W_{SGNS}$, we set negative sampling as 5. For both, we set a context window size of 2 (taking 5 words as context) and embedding dimension $d = 500$.

5 Results

5.1 Distribution of Eigenvector Elements

From Figure 1, we can see that eigenvectors corresponding to the larger eigenvalues such as $u^1$ or $u^2$ clearly deviate from a Gaussian distribution, and so do $u^{100}$ and $u^{500}$, but less. This shows us that the eigenvectors are not random and contain meaningful correlations. It is expected to see such pattern because these vectors are the principal eigenvectors.

On a more interesting note, $u^1$ not only significantly deviates from a normal distribution, but also has only non-zero negative values as its elements, and no other eigenvectors have shown this behavior. This suggests that this particular eigenvector could represent a common bias that affects all “words”, as it captured the effect of news outbreaks for stock prices in [19]. We revisit the interpretation of this observation in Section 6.1.

5.2 Inverse Participation Ratio

Figure 2 illustrates the IPR of $u^k$ plotted against the corresponding eigenvalue $\lambda^k$, and vice versa for $v^k$. From the plot, we can clearly see that the eigenvectors of $W_{SVD}$ have approximately 10x higher IPR values than those of $W_{SGNS}$, meaning that the vectors are much sparser for $W_{SVD}$.

From Figure 2a, we can see that the largest eigenvector has the smallest IPR of 0.000006, and the reciprocal $1/\text{IPR}$ divided by $|V|$, yields 48%, while the same for the largest $I^k$ gave around 4.7%. The mean value of $1/\text{IPR}$ divided by $|V|$, across all eigenvectors was 27.5% indicating that there exists

\[http://bitbucket.org/omerlevy/hyperwords\]
some sparse structure within the eigenvectors of $W^{\text{SVD}}$. On the other hand, Figure 2b shows that mean for $v^4$ was around 36%, meaning that column vectors of $W^{\text{SGNS}}$ are generally denser and less structured. Such discrepancy in structural sparsity motivates us to analyze the eigenvectors of $W^{\text{SVD}}$ in depth.

6 Analysis and Discussion

6.1 Column Space Analysis

Based on the results of previous sections, we further examine the top elements of the eigenvectors by sorting their absolute values in decreasing order. Table 1 shows interesting results as the significant dimensions or their corresponding “words” of each eigenvector, in general, form semantically or syntactically coherent groups. For instance, $u^{14}$ groups French words together and $u^{121}$ shows baseball related words. Some words from $u^{121}$ initially seem irrelevant to baseball. However, “buehrle” is a baseball player, “rbis” stand for “Run Batted Ins”, and “astros” is a baseball team name.
Table 2: Top participants of eigenvectors (dimensions with highest magnitudes) of $W_{SVD}$ with largest IPR values (highest sparsity) in decreasing order.

<table>
<thead>
<tr>
<th>$u^{42}$</th>
<th>$u^{50}$</th>
<th>$u^{14}$</th>
<th>$u^{101}$</th>
<th>$u^{75}$</th>
<th>$u^{121}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>stani</td>
<td>bandeira</td>
<td>famille</td>
<td>seon</td>
<td>bucharest</td>
<td>jays</td>
</tr>
<tr>
<td>kne</td>
<td>concei</td>
<td>vrier</td>
<td>hyeon</td>
<td>lescu</td>
<td>strikeouts</td>
</tr>
<tr>
<td>vukovi</td>
<td>nio</td>
<td>autour</td>
<td>seung</td>
<td>ional</td>
<td>halladay</td>
</tr>
<tr>
<td>kovi</td>
<td>jardim</td>
<td>naissance</td>
<td>seong</td>
<td>ntul</td>
<td>hitters</td>
</tr>
<tr>
<td>vuk</td>
<td>velho</td>
<td>rique</td>
<td>hwang</td>
<td>napoca</td>
<td>buehrle</td>
</tr>
<tr>
<td>kova</td>
<td>visconde</td>
<td>diteur</td>
<td>choi</td>
<td>editura</td>
<td>batters</td>
</tr>
<tr>
<td>popovi</td>
<td>pessoa</td>
<td>octobre</td>
<td>sik</td>
<td>iancu</td>
<td>pitching</td>
</tr>
<tr>
<td>inovi</td>
<td>domingos</td>
<td>chambre</td>
<td>kyung</td>
<td>romanian</td>
<td>phillies</td>
</tr>
<tr>
<td>uro</td>
<td>pinheiro</td>
<td>lettre</td>
<td>kwang</td>
<td>mihai</td>
<td>rbis</td>
</tr>
<tr>
<td>mileti</td>
<td>branco</td>
<td>campagne</td>
<td>jeong</td>
<td>institut</td>
<td>astros</td>
</tr>
<tr>
<td>novakovi</td>
<td>nicio</td>
<td>jeune</td>
<td>yeong</td>
<td>tilor</td>
<td>diamondbacks</td>
</tr>
<tr>
<td>anovi</td>
<td>trindade</td>
<td>jours</td>
<td>taek</td>
<td>traien</td>
<td>homers</td>
</tr>
<tr>
<td>filipovi</td>
<td>carmo</td>
<td>septembre</td>
<td>gyeong</td>
<td>ianu</td>
<td>hitless</td>
</tr>
<tr>
<td>petrov</td>
<td>cio</td>
<td>enface</td>
<td>seok</td>
<td>pentru</td>
<td>orioles</td>
</tr>
<tr>
<td>radi</td>
<td>ssimo</td>
<td>plon</td>
<td>chose</td>
<td>tefan</td>
<td>podsednik</td>
</tr>
<tr>
<td>evi</td>
<td>penha</td>
<td>affaire</td>
<td>ryong</td>
<td>muzeul</td>
<td>baserunners</td>
</tr>
<tr>
<td>knez</td>
<td>paulo</td>
<td>cembre</td>
<td>hee</td>
<td>gheorghe</td>
<td>hitter</td>
</tr>
<tr>
<td>martinovi</td>
<td>cavallo</td>
<td>royaume</td>
<td>gwang</td>
<td>bucreesti</td>
<td>sox</td>
</tr>
<tr>
<td>vuka</td>
<td>marinho</td>
<td>propos</td>
<td>cheol</td>
<td>biserica</td>
<td>petitte</td>
</tr>
<tr>
<td>veljko</td>
<td>neves</td>
<td>juin</td>
<td>myeong</td>
<td>craiova</td>
<td>vizquel</td>
</tr>
</tbody>
</table>

Table 3: Top participants of the salient columns of the word vector for “airport.”

<table>
<thead>
<tr>
<th>$W_{SVD}$</th>
<th>$W_{SGNS}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>located</td>
<td>garabel</td>
</tr>
<tr>
<td>near</td>
<td>hagopian</td>
</tr>
<tr>
<td>connecting</td>
<td>activist</td>
</tr>
<tr>
<td>situated</td>
<td>abrahamyan</td>
</tr>
<tr>
<td>connects</td>
<td>fumicino</td>
</tr>
<tr>
<td>bandeira</td>
<td>marsh</td>
</tr>
<tr>
<td>trindade</td>
<td>voices</td>
</tr>
<tr>
<td>penha</td>
<td>papazian</td>
</tr>
<tr>
<td>concei</td>
<td>easyjet</td>
</tr>
<tr>
<td>velho</td>
<td>erden</td>
</tr>
<tr>
<td></td>
<td>documentaries</td>
</tr>
</tbody>
</table>

from Houston. Meanwhile, the words grouped in $u^1$, the largest eigenvector, could explain the bias we mentioned in Section 5.1. The significant participants tend to be strong transition words that are used often for dramatic effects, such as “importantly” or “crucially”. Evidently, these words increase the intensity of the context.

Moreover, while it was originally hypothesized that the largest principal eigenvectors would capture some semantic relationship, the 121th vector $u^{121}$ show surprisingly focused and narrow semantic grouping related to baseball. Further investigation reveals that $u^{121}$ has one of the highest IPR values, hence being one of the most sparse vectors. We verify similar trends in other eigenvectors with high IPR values as shown in Table 2. An interesting pattern arises here, in which the sparser eigenvectors tend to capture more distinct and rare features such as foreign names or languages, or topics like baseball.

Furthermore, we compare the column space analysis on $W_{SVD}$ and $W_{SGNS}$. Consider the word vector $w_{airport}$ for the word “airport.” We choose the salient dimensions, which are the largest elements, of $w_{airport}$, and investigate the significant elements of those chosen dimensions (columns). Table 3 shows that the columns from $W_{SVD}$ display semantic coherence while those from $W_{SGNS}$ seem random. $u^{43}$ groups words that are related to the location of the airports. For example, one could
Figure 3: Comparison of the representations of four baseball related words in $W^{\text{SVD}}$ and $W^{\text{SGNS}}$, where the rows are embedding dimensions. It is clear that $W^{\text{SVD}}$ shows similar representations.

6.2 Word Embedding Dimensions as Interpretable Features

Sections 5.1 and 5.2 revealed that the eigenvectors contain genuine correlation and structure in the column space. We further show in Section 6.1 that semantically coherent words form groups of significant participants in each eigenvector. Now we can answer the questions we asked earlier.

**What is the meaning of high and low values in the columns of $W$?** If word vector $w$ from $W^{\text{SVD}}$ has a high absolute value in column $k$, it means that the word is relevant to the semantic group formed in $u^k$. For example, the words from Figure 3a have highest values in column $k = 121$, in which $u^{121}$ represents a semantic group related to baseball, as shown in Table 1.

**How can we interpret the dimensions of word vectors?** The answer to this question follows naturally. As the salient dimensions represent relevant semantic groups, we can view the dimensions of $w$ as semantic features. This view is in line with the Topic Modeling literature, in which words and documents are clustered into distinct latent topics. Hence, we can also see the word embedding dimensions as latent topics that can be interpretable.

It can be easily seen from Figure 3b that similar words do not show any interpretable similarity in their $W^{\text{SGNS}}$ representations, despite being nearest neighbors in the row space. On the other hand, it is very clear from Figure 3a that similar words have similar representations, or feature vectors. We thus empirically verify that the dimensions of the row vectors can be viewed as semantic or syntactic features. Finally, the structural sparsity discovered with the IPR is further confirmed by contrasting Figures 3a and 3b. It is clearly visible that the the vectors from SVD are much sparser than from SGNS.

**Effect of Eigenvalue Weighting:** As mentioned in Section 5, weighting with the eigenvalues essentially scales each feature column by the corresponding eigenvalues. Such process can be viewed as simply incorporating a prior, and does not hurt the interpretability. However, as [18] showed that eigenvalue weighting decreases the performance of downstream NLP tasks, we can assume that either
the prior is wrong, or too strong. In fact, in many cases, the largest eigenvalues are often order of magnitude larger than others, which can explain why not weighting the word embeddings with their corresponding eigenvalues would work better.

7 Conclusion

In this work, we analyzed the eigenvectors, or the column space, of the word embeddings obtained from the Singular Value Decomposition of PPMI matrix. We revealed that the significant participants of the eigenvectors form semantically coherent groups, allowing us to view each word vector as an interpretable feature vector composed of semantic groups. These results can be very useful in error analysis in downstream NLP tasks, or cherry-picking useful feature dimensions to easily create compressed and efficient task-specific embeddings. Future work will proceed in this direction on applying interpretability to practical usage.

References


