Word2Vec is only a special case of Kernel Correspondence Analysis and Kernels for Natural Language Processing

Hirotaka Niitsuma
Department of Computer Science
Okayama University
Okayama, Japan
niitsuma@cs.okayama-u.ac.jp

May 18, 2016

Abstract

We show Correspondence Analysis (CA) is equivalent to defining Gini-index with appropriate scaled one-hot encoding. Using this relation, we introduce non-linear kernel extension of CA. The extended CA gives well-known analysis for categorical data (CD) and natural language processing by specializing kernels. For example, our formulation can give G-test, skip-gram with negative-sampling (SGNS), and GloVe as a special case. We introduce two kernels for natural language processing based on our formulation. First is a stop word (SW) kernel. Second is word similarity (WS) kernel. The SW kernel is the system introducing appropriate weights for SW. The WS kernel enables to use WS test data as training data for vector space representations of words. We show these kernels enhances accuracy when training data is not sufficiently large.

1 Introduction

Principal Component Analysis (PCA) and Canonical Correlation Analysis (CCA) are the well-known multivariate data reduction techniques. These two methods are an analysis of a correlation or variance-covariance matrix of a multivariate quantitative data set. CA is an analysis of CD.

CA is also a statistical visualization method for picturing the associations between the levels of a two-way contingency table. To illustrate CA, consider the contingency table shown in Table 1. Table 1 is well-known as Fisher’s data on colors of eyes and hair of the people in Caithness, Scotland. CA of this data yields the graphical display shown in Figure 1. In Figure 1, we can see the correspondence between colors of eyes and hair.
PCA is the eigenvalue decomposition of a covariance matrix. CCA is the singular value decomposition (SVD) based on the matrix representing a correlation with modifying the average and the inhomogeneity. CA can be regarded as SVD of the covariance of CD with modifying the average and the inhomogeneity like CCA. Gini-index is another way dealing with the variance of CD. Okada [11, 10] showed defining Gini-index using one-hot encoding on appropriately rotated space gives reasonable values for the covariance. In this research, we show the rotated Gini-index with modifying inhomogeneity is equivalent to CA.

Picca et al. [13] introduced non-linear kernel extension of CA. We [9] introduced non-linear kernel extension of the rotated Gini-index. The equivalence between CA and the modified Gini-index can define more general non-linear kernel extension for CD. Let us call this non-linear kernel extension as kernel Correspondence Analysis (KCA). KCA gives well-known analysis for CD and natural language processing by specializing kernels. For example, our KCA can give G-test, skip-gram with negative-sampling (SGNS) [8, 6], and GloVe [12] as a special case.

We apply KCA to natural language processing. Especially we focus on vector representations of words. Mikolov et al. [8] introduced dense vector representations of words referred as word2vec. The vector representations give meaning subtraction or addition operations of the vectors. For example, the analogy king is to queen as man is to woman, is encoded in the vector space by the vector equation: king − queen = man − woman. Note that sometimes CA also has the similar property. For example we can see similar property in Figure 1.

Levy et al. [6] showed skip-gram with negative-sampling (SGNS), which is one of the word2vec, can be regarded as analysis of the two-way contingency table in which the row represents a word, and the column represents context. [6] showed SGNS is the matrix factorization of the pointwise mutual information (PMI) of the contingency table. PMI can be regard as one of the log scale representation of a contingency table. We formulate this mechanism within KCA by giving appropriate kernels.

2 Correspondence Analysis

|        | x<sup>fair</sup> | x<sup>red</sup> | x<sup>medium</sup> | x<sup>dark</sup> | x<sup>black</sup> |
|--------|-----------------|-----------------|-------------------|-----------------|------------------|
| blue   | 326             | 38              | 241               | 110             | 3                |
| light  | 688             | 116             | 584               | 188             | 4                |
| medium | 343             | 84              | 909               | 412             | 26               |
| dark   | 98              | 48              | 403               | 681             | 85               |
CA is generalized singular value decomposition (GSVD) based on a contingency table for two categorical variables so that

\[ USV^t = \Xi = \frac{N}{n} - \mathbf{r}^t \mathbf{c} / n^2, \quad U^t D(\mathbf{r})^{-1} U = V^t D(\mathbf{c})^{-1} V = \mathbf{E} \]  

where \( N \) is an \( n^r \times n^c \) contingency table, whose entries \( n_{ij} \) give the frequency with which row categorical variable \( x^r = i \) occurs together with column categorical variable \( x^c = j \). \( x^r \in \{ x^r_1, x^r_2, ..., x^r_{n^r} \} \) is the categorical variable representing the row side of the contingency table \( N \). \( x^c \in \{ x^c_1, x^c_2, ..., x^c_{n^c} \} \) is the categorical variable representing the column side. \( \mathbf{r} = N \mathbf{1} \) denote the vector of row marginals, \( \mathbf{c} = N^\mathbf{1} \) is the vector of column marginals and \( n = \mathbf{1}^t \mathbf{c} = \mathbf{1}^t \mathbf{r} \) is the total number of observations, where \( \mathbf{1} = (1, 1, 1, ...)^t \). \( \mathbf{E} \) is an identity matrix. \( D(\mathbf{v}) \) is the diagonal matrix whose diagonal entries are the components of vector \( \mathbf{v} \). \( USV^t = \Xi \) is GSVD of \( \Xi \) so that \( U^t D(\mathbf{r})^{-1} U = V^t D(\mathbf{c})^{-1} V = \mathbf{E} \).

For example, let us consider the contingency table shown at Table 1, which is well known as Fisher's data [4] on colors of eyes and hair of the people in Caithness, Scotland. The table represents the joint population distribution of the categorical variable for eye color \( x^{eye} \in \{ \text{fair, red, medium, dark, black} \} \) and the categorical variable for hair color \( x^{hair} \in \{ \text{blue, light, medium, dark} \} \). For this data, \( \mathbf{r} = (718, 1580, 1774, 1315)^t \), \( \mathbf{c} = (1455, 286, 2137, 1391, 118)^t \), and \( n = 5387 \). CA using Fisher’s data is GSVD of

\[
\frac{1}{n} \mathbf{N} - \frac{1}{n^2} \mathbf{r} \mathbf{c}^t = \frac{1}{5387} \begin{bmatrix} 326 & 38 & 241 & 110 & 3 \\ 688 & 116 & 584 & 188 & 4 \\ 343 & 84 & 909 & 412 & 26 \\ 98 & 48 & 403 & 681 & 85 \end{bmatrix} + \frac{1}{5387^2} \begin{bmatrix} 718 \\ 1580 \\ 1774 \\ 1315 \end{bmatrix} \begin{bmatrix} 1455 \\ 286 \\ 2137 \\ 1391 \\ 118 \end{bmatrix}^t.
\]

(2)

Usually the decomposed matrix \( U \) and \( V \) is used for PCA for these categorical variables.

Figure 1: Visualizing Fisher’s data.
3  Gini’s definition of Variance and its Extension

Note that a contingency table can be constructed by means of the matrix product of two indicator matrices:

\[ N = H^r H^c \]  

(3)

where

\[ H^r = [\mathbf{e}^r(x^r(a_1)), \mathbf{e}^r(x^r(a_2)), ..., \mathbf{e}^r(x^r(a_n))]^t \]
\[ H^c = [\mathbf{e}^c(x^c(a_1)), \mathbf{e}^c(x^c(a_2)), ..., \mathbf{e}^c(x^c(a_n))]^t. \]  

(4)

\( H^r \) is the \( n \times n^r \) indicator matrix of \( x^r \). \( \mathbf{e}^r(x^r) \) is one-hot encoding of \( x^r \). \( a_1, ..., a_n \) represent the observations. \( x^r(a_b) \) is the value of \( x^r \) of \( b \)-th observation. \( H^c \) is an \( n \times n^c \) indicator matrix of \( x^c \). \( \mathbf{e}^c(x^c) \) is one-hot encoding of \( x^c \). \( x^c(a_b) \) is the value of \( x^c \) of \( b \)-th observation.

For example, the one-hot encodings and indicator matrix of \( x^\text{eye} \) in Fisher’s data are

\[ \mathbf{e}^\text{eye}(\text{blue}) = (1, 0, 0, 0)^t \]
\[ \mathbf{e}^\text{eye}(\text{light}) = (0, 1, 0, 0)^t \]
\[ \mathbf{e}^\text{eye}(\text{medium}) = (0, 0, 1, 0)^t \]
\[ \mathbf{e}^\text{eye}(\text{dark}) = (0, 0, 0, 1)^t \]

\( H^\text{eye} \)

\[ H^\text{eye} = \begin{bmatrix} \mathbf{e}^\text{eye}(\text{eye color of 1st person})^t \\ \vdots \\ 0, 0, 1, 0 \\ \vdots \\ \mathbf{e}^\text{eye}(\text{eye color of nth person})^t \end{bmatrix} \]  

(5)

\( H^\text{hair} \) can be constructed in the same manner. The contingency table of Fisher’s data is

\[ N = H^{\text{eye}} H^{\text{hair}}. \]  

(6)

Based on each observation, we can define Gini-index:

\[ \sigma^2(x) = \frac{1}{2n^2} \sum_{a=1}^{n} \sum_{b=1}^{n} (x(a) - x(b))^2 \]  

(7)

where

\[ (x(a) - x(b))^2 = \begin{cases} 0 & \text{if } x(a) = x(b) \\ 1 & \text{if } x(a) \neq x(b) \end{cases} \]  

(8)

Gini index is the variance of a categorical variable\[5\]. For example, \( \sigma^2(x^\text{eye}) \) using Fisher’s data is

\[ \sigma^2(x^\text{eye}) = \frac{1}{2 \cdot 5387^2} \sum_{a=1}^{5387} \sum_{b=1}^{5387} (\text{eye color of a th person}) - (\text{eye color of b th person}). \]

A simple extension of this definition to the covariance \( \sigma^2(x^r, x^c) \) by replacing \( (x(a) - x(b))^2 \) to \( (x^r(a) - x^r(b))(x^c(a) - x^c(b)) \) does not give reasonable values
for the covariance $\sigma^2(x^r, x^c)$. Okada showed using rotated one-hot encoding gives reasonable values for the covariance of CD:

$$\sigma^2(x^r, x^c) = \maximize_R \frac{1}{2n^2} \sum_{a=1}^{n} \sum_{b=1}^{n} (e^r(x^r(a)) - e^r(x^r(b)))^t R(e^c(x^c(a)) - e^c(x^c(b)))$$

subject to $R^t R = E$  \hspace{1cm} (9)

where $R$ is a rotation matrix so that maximize the covariance. We can rewrite Eq. (9) using $\Xi$ which is used in CA.

$$\frac{1}{2n^2} \sum_{a=1}^{n} \sum_{b=1}^{n} (e^r(x^r(a)) - e^r(x^r(b)))^t R(e^c(x^c(a)) - e^c(x^c(b)))$$

$$= \frac{1}{2n^2} \text{tr}(R^t \sum_{a=1}^{n} \sum_{b=1}^{n} (e^r(x^r(a)) - e^r(x^r(b)))(e^c(x^c(a)) - e^c(x^c(b)))^t)$$

$$= \text{tr}(R^t (\frac{1}{n} \sum_{a=1}^{n} e^r(x^r(a)) e^c(x^c(a))^t - \frac{1}{n^2} \sum_{a=1}^{n} e^r(x^r(a)) \sum_{b=1}^{n} e^c(x^c(b))^t))$$

$$= \text{tr}(R^t (N/n - r c^t/n^2)) = \text{tr}(R^t \Xi)$$  \hspace{1cm} (10)

We can solve the maximization problem by differentiating the following Lagrangian.

$$\mathcal{L} = \text{tr}(R^t \Xi) - \text{tr}(\Lambda^t (R^t R - E))$$  \hspace{1cm} (11)

where $\Lambda$ is a Lagrange multiplier. The conditions for this Lagrangian $\mathcal{L}$ to be stationary with respect to $R$ is $R^t \Xi = (\Lambda + \Lambda^t)$. This shows $(R^t \Xi)$ must be symmetric matrix. We can give a solution to this maximization problem using the following SVD.

$$U^t S V^t = \Xi$$  \hspace{1cm} (12)

We can see $R = U^t V^t$ is satisfying all required conditions and conclude $R = U^t V^t$ is the solution of the problem (9). Recall CA is the GSVD of $\Xi$. The GSVD $USV^t = \Xi$ can be computed by using the SVD $U^t S V^t = \Xi = D(r)^{-1/2} \Xi D(c)^{-1/2}$ as the following.

$$U = D(r)^{1/2} U'$$ ,  \hspace{1cm} $V = D(c)^{1/2} V'$  \hspace{1cm} (13)

Substituting this relation to Eq. (9) gives

$$\maximize_R \text{tr}(RD(c)^{-1} \Xi D(r)^{-1})$$

subject to $D(c)^{-1} R D(r)^{-1} R = E$  \hspace{1cm} (14)

The solution of this optimization problem is $R = UV^t$. We can say CA is equivalent to the problem (14).

Based on this relation, let us introduce scaled one-hot encoding as the following.

$$f^r(x^r) = D(r)^{-1} e^r(x^r)$$ ,  \hspace{1cm} $f^c(x^c) = D(c)^{-1} e^c(x^c)$  \hspace{1cm} (15)
Note that

\[
[F(x_1^r), F(x_2^r), \ldots, F(x_n^r)] = D(r)^{-1} [e^r(x_1^r), e^r(x_2^r), \ldots, e^r(x_n^r)] = D(r)^{-1} E = D(r)^{-1}
\]

Substituting this relation to the problem (14) gives

\[
\text{maximize}_{R} \frac{1}{2n^2} \sum_{a,b} (F(x^r(a)) - F(x^r(b)))^t R(F(x^r(a)) - F(x^r(b)))
\]

subject to \(R[f^r(x_1^r), f^r(x_2^r), \ldots] = E\) (17)

We can see this problem defines the rotated Gini-index using the scaled one-hot encoding. And this generalized Gini-index is equivalent to CA.

4 Non-linear extension

This section attempts to generalize CA as possible as we can generalize within keeping the relation between Gini-index and CA. Let us introduce non-linear mapping \(\Phi^r, \Phi^c\) and operators on the non-linear mapped spaces \(\ominus^r, \ominus^c, \ominus, \oplus\). Rewriting the problem (17) using this non-linear extension gives:

\[
\text{maximize}_{R} \frac{1}{2n^2} \sum_{a,b} (\Phi^r(F^r(x^r(a))) \ominus^r \Phi^r(F^r(x^r(b))))^t R(\Phi^r(F^r(x^r(a))) \ominus^c \Phi^c(F^c(x^c(b))))
\]

subject to \(R[f^r(x_1^r), f^r(x_2^r), \ldots] = E\) (18)

Consider expanding this expression with the following rules:

\[
(Y_1 \ominus^y Y_2)X^t = Y_1X^t \ominus Y_2X^t \quad (19)
\]

\[
Y(X_1 \ominus^x X_1)^t = YX_1^t \ominus YX_1^t \quad (20)
\]

\[
X \ominus (Y \ominus Z) = (X \ominus Z) \ominus Y \quad (21)
\]

\[
(X \ominus Y) \ominus Z = (X \ominus Z) \ominus Y \quad (22)
\]

\[
(X \ominus Y) \ominus Z = X \ominus (Z \ominus Y) \quad (23)
\]

\[
X \ominus (Y \ominus Z) = (X \ominus Y) \ominus Z \quad (24)
\]

\[
\theta_{ij} = \Phi^r(F^r(x_i^r))R\Phi^c(F^c(x_j^c)) \quad (25)
\]

\[
\Theta = [\theta_{ij}] \quad (26)
\]

Let us consider the case, how to evaluate an expression including the operators \(\ominus, \ominus^x, \ominus^y\) is not defined, and we need to move these operators outside paraneces. In order to move the operators outside paraneces, rules (19) to (24) can be used. In these rules, rewriting left-hand side to right-hand side moves
Table 2: Relations between known methods and kernel specializations.

| Name                | $K^r$            | $K^c$            | $X \ominus Y$ | $X \oplus Y$ |
|---------------------|------------------|------------------|----------------|--------------|
| Linear CA           | $D(r)^{-1}$      | $D(c)^{-1}$      | $X - Y$        | $X + Y$      |
| Gini-index          | $E$              | $E$              | $X - Y$        | $X + Y$      |
| G-test              | $E$              | $E$              | $(\log X - \log Y)$ | $X + Y$      |
| SGNS                | $E$              | $E$              | $(\log X - \log Y - \log k)$ | $X + Y$      |
| GloVe               | $E$              | $E$              | $(\log X - \log Y + b^u + b^c)$ | $X + Y$      |
| kernel PCA for CD   | $e^{\alpha(x^r) - e^c(x^r)^2}$ | $E$              | $X - Y$        | $X + Y$      |

the operators outside paraneces. Applying these rewriting rules to problem (18) gives:

\[
\begin{align*}
\text{maximize} & \quad \frac{1}{2n^2} \text{tr}(R^t(2nN\oplus 2\Upsilon)) \\
\text{subject to} & \quad R^t\Theta = E \quad (27)
\end{align*}
\]

where

\[
N\oplus = \sum_{a=1}^n \Phi^r(f^r(x^r(a)))\Phi^c(f^c(x^c(a)))^t
\]

\[
2\Upsilon\oplus = \sum_{a=1}^n \sum_{b=1}^n \Phi^r(f^r(x^r(a)))\Phi^c(f^c(x^c(b)))^t \quad (28)
\]

When $\oplus = +$, and introducing kernel matrices $K^r$ and $K^c$ so as to be

\[
\Theta = [\theta_{ij}] = K^r R K^c
\]

gives

\[
\begin{align*}
\text{maximize} & \quad \text{tr}(R^tK^r(N/n \ominus rc^t/n^2)K^c) \\
\text{subject to} & \quad R^tK^r R K^c = E. \quad (31)
\end{align*}
\]

Specifying the operators and kernel matrices provides various well known analysis for CD and natural language processing. Table 2 shows the relation between the specification and the known methods. Linear CA is a simple CA without the non-linear extension. Levy et al. [7] showed GloVe is matrix factorization of sifted PMI. Based on that discussion, we can regard GloVe is the one of the KCA as showed in Table 2. When some element of a contingency table is zero: $n_{ij} = 0$ then PMI = $-\infty$. G-test for contingency tables can avoid this problem. Let us call SVD of $\Xi$ when $X \ominus Y = X(\log X - \log Y)$ as G-test in this study.
In this section, we focus on vector representations of words. Let us consider the following contingency table

\[ N = [n_{ij}] = \#(w_i, c_j) \] (32)

where \#(w, c) is the number of times the word \( w \) appears in the context \( c \). Using this contingency table, we can formulate SGNS as one of the specialized KCA as showed in Table 2. Linear CA and G-test also provide vector representations of words using the contingency table.

KCA can introduce tunable weights for stop words (SW) using the following kernel:

\[ K^r = D(\text{StopWordVector}(\alpha w))D(r)^{-1} \quad K^c = D(\text{StopWordVector}(\alpha c))D(c)^{-1} \] (33)

where

\[ \text{StopWordVector}(\alpha) = (\ldots, 1 + \alpha \delta_{i \in \text{StopWords}}, \ldots) \] (34)

is the vector in which the weight \( \alpha \) is added when \( i \)-th word is SW. Let us call this mechanism as stop word kernel.

When we have data set about similarity scores between two words, the scores can be encoded into the vector representations of words using the our kernel mechanism. For example,[3]

\[ \text{score(movie, theater)} = 7.92 \]
\[ \text{score(movie, star)} = 7.38 \]
\[ \text{score(movie, popcorn)} = 6.19 \]

We can encode such scores using defining operators \( \ominus^r \) and \( \ominus^c \) in Eq. (18) as the following.

\[
\Phi^r(f^r(x^r_a)) \ominus^r \Phi^r(f^r(x^r_b)) = (f^r(x^r_a) - f^r(x^r_b))\gamma^r(x_a, x_b) \\
\Phi^c(f^c(x^c_a)) \ominus^c \Phi^c(f^c(x^c_b)) = (f^c(x^c_a) - f^c(x^c_b))\gamma^c(x_a, x_b) 
\] (35)

where

\[
\gamma^r_{ij} = \alpha^r \text{score}(i, j) + \beta^r \quad \gamma^c_{ij} = \alpha^c \text{score}(i, j) + \beta^c 
\] (36)

\( \alpha^r, \beta^r, \alpha^c, \beta^c \) are tunable parameters. When \( \ominus = - \) and \( \oplus = + \), the problem (18) is

\[
\text{maximize} \quad \frac{1}{n^2} \text{tr}(R^t D(r)^{-1} (N \circ (\Gamma^r N \Gamma^c) - (\Gamma^r N) \circ (N \Gamma^c)) D(c)^{-1}) \\
\text{subject to} \quad R^t D(r)^{-1} R D(c)^{-1} = 1 \\
r = (\Gamma^r N) \circ (N \Gamma^c)1 \\
c = ((\Gamma^r N) \circ (N \Gamma^c))^t 1 \\
\Gamma^r = [\gamma^r_{ij}] \\
\Gamma^c = [\gamma^c_{ij}] 
\] (37)
where $\circ$ is Hadamard product. Let us call this mechanism as word similarity kernel.

6 Experiments

This section shows experimental evaluations of vector representations of words based on our proposed system. To calculate the contingency table $\mathbf{N} = [n_{ij}] = [\#(w_i, c_j)]$, we use the program code used in \cite{1}. For SW kernel, we use stop words set in NLTK\cite{1}. For WS, we use the score of Bruni et al.’s MEN dataset\cite{2}.

First 20% data of Text8 corpus\cite{2} is used as the training text data. The stop words set and word similarity data is too small compared to whole Text8 corpus. Thus stop word kernel and word similarity kernel have small effect when using whole Text8 corpus.

For the evaluation, we use WordSim3533 dataset\cite{3}, Bruni et al.’s MEN dataset\cite{2}, and Radinsky et al.’s Mechanical Turk dataset\cite{14}.

The word vectors are evaluated by comparing ranking their cosine similarities and the ranking in the test data set. The rankings are compared using Spearman’s $\sigma$. We show also the result of SGNS for the comparison. Table 3 is the evaluation results. Linear CA provides 4 types vector representations: $U$, $V$, $U'$, $V'$. We show the results of these 4 representations. We also show the results of G-test. In G-test, $U = U'$ and $V = V'$. Thus we omit results for $U'$ and $V'$. The results for Linear CA and G-test with SW kernel and WS kernel is showed as “with SW” or “with WS” in the Table.

In spite of Linear CA and G-test have no tunable parameter, there results are comparable to SGNS. In most case, SW kernel enhances accuracy. WS kernel slightly enhances accuracy in some cases.

7 Conclusion

We show Liner Correspondence Analysis(CA) is equivalent to defining Gini-index with rotated and scaled one-hot encoding. And we attempts to generalize CA as possible as we can generalize within keeping the relation between Gini-index and CA. Based on the non-linear generalization of CA, Kernel Correspondence Analysis(KCA) is introduced. KCA gives various known analysis for categorical data and natural language processing by specializing kernels. For example, KCA can give G-test, skip-gram with negative-sampling(SGNS), and GloVe as a special case. We introduce two kernels for natural language processing based on KCA. The proposed mechanism is evaluated by applying the problem of vector representations of words. In spite of Linear CA and G-test have no tunable parameter, there results are comparable to SGNS. Also, we show kernels with tunable parameters can enhance accuracy.

\textsuperscript{1}https://bitbucket.org/omerlevy/hyperwords
\textsuperscript{2}http://mattmahoney.net/dc/text8.zip
### Table 3: Comparing word similarity.

| Name                     | WordSim3533 | Bruni | Randinsky |
|--------------------------|-------------|-------|-----------|
| SGNS                     | 0.192       | 0.061 | 0.314     |
| Linear CA V              | 0.189       | 0.081 | 0.201     |
| Linear CA V′             | 0.161       | 0.101 | 0.194     |
| Linear CA V″              | 0.179       | 0.084 | 0.365     |
| Linear CA with SW V      | 0.198       | 0.088 | 0.212     |
| Linear CA with SW V′     | 0.182       | 0.102 | 0.172     |
| Linear CA with SW V″     | 0.198       | 0.094 | 0.383     |
| Linear CA with WS V      | 0.189       | 0.084 | 0.201     |
| Linear CA with WS V′     | 0.213       | 0.074 | 0.407     |
| Linear CA with WS V″     | 0.161       | 0.102 | 0.194     |
| Linear CA with WS V′′    | 0.179       | 0.084 | 0.365     |
| G-test U                 | 0.139       | 0.074 | 0.156     |
| G-test V                 | 0.169       | 0.051 | 0.365     |
| G-test with SW U         | 0.178       | 0.078 | 0.179     |
| G-test with SW V         | 0.186       | 0.065 | 0.369     |
| G-test with WS U         | 0.139       | 0.074 | 0.157     |
| G-test with WS V         | 0.169       | 0.053 | 0.365     |

### References

[1] S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python*. O’Reilly Media, 2009.

[2] E. Bruni, G. Boleda, M. Baroni, and N. K. Tran. Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 136–145. Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[3] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing search in context: The concept revisited. *ACM Trans. Inf. Syst.*, 20(1):116–131, Jan. 2002.

[4] R. A. Fisher. The precision of discriminant functions. *Annals of Eugenics (London)*, 10:422–429, 1940.

[5] C. Gini. Variability and mutability, contribution to the study of statistical distributions and relations. studi economico-giuridici della r. universita de cagliari (1912). reviewed in: Light, r.j., margolin, b.h.: An analysis of variance for categorical data. *J. American Statistical Association*, 66:534–544, 1971.
Table 4: Whole text8 corpus used

| Name          | WordSim3533 | Bruni | Randinsky |
|---------------|-------------|-------|-----------|
| SVD[7]        | 0.392       | 0.087 | 0.499     |
| SGNS[7]       | 0.443       | 0.098 | 0.551     |
| Linear CA F   | 0.364       | 0.121 | 0.291     |
| Linear CA G   | 0.415       | 0.118 | 0.449     |
| Linear CA U   | 0.362       | 0.140 | 0.313     |
| Linear CA V   | 0.413       | 0.138 | 0.484     |
| G-test U      | 0.260       | 0.128 | 0.268     |
| G-test V      | 0.323       | 0.081 | 0.404     |

[6] O. Levy and Y. Goldberg. Neural word embedding as implicit matrix factorization. In Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada, pages 2177–2185, 2014.

[7] O. Levy, Y. Goldberg, and I. Dagan. Improving distributional similarity with lessons learned from word embeddings. TACL, 3:211–225, 2015.

[8] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, editors, Advances in Neural Information Processing Systems 26, pages 3111–3119. Curran Associates, Inc., 2013.

[9] H. Niitsuma and T. Okada. Kernel pca for categorical data. IEICE technical report. Artificial intelligence and knowledge-based processing, 103(305):13–17, sep 2003.

[10] H. Niitsuma and T. Okada. Covariance and PCA for categorical variables. In Advances in Knowledge Discovery and Data Mining, 9th Pacific-Asia Conference, PAKDD 2005, Hanoi, Vietnam, May 18-20, 2005, Proceedings, pages 523–528, 2005.

[11] T. Okada. A note on covariances for categorical data. In K. Leung, L. Chan, and H. Meng, editors, Intelligent Data Engineering and Automated Learning - IDEAL 2000, 2000.

[12] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In Empirical Methods in Natural Language Processing (EMNLP), pages 1532–1543, 2014.

[13] D. Picca, B. Curdy, and F. Bavaud. Non-linear correspondence analysis in text retrieval: A kernel view. In Proceedings of JADT, 2006.

[14] K. Radinsky, E. Agichtein, E. Gabrilovich, and S. Markovitch. A word at a time: Computing word relatedness using temporal semantic analysis.
