Natural Language Engineering (2023), 1–31
doi:10.1017/S1351324923000244

CAMBRIDGE UNIVERSITY PRESS

ARTICLE

A comparison of latent semantic analysis and correspondence analysis of document-term matrices

Qianqian Qi[1], David J. Hessen[1], Tejaswini Deoskar[2] and Peter G. M. van der Heijden[1,3]

[1]Department of Methodology and Statistics, Faculty of Social Sciences, Utrecht University, Utrecht, the Netherlands,[2]Department of Languages, Literature, and Communication, Faculty of Humanities, Utrecht University, Utrecht, the Netherlands, and [3]Southampton Statistical Sciences Research Institute, University of Southampton, Highfield, Southampton, UK
Corresponding author: Qianqian Qi; Email: q.qi@uu.nl

(Received 12 July 2022; revised 23 March 2023; accepted 17 April 2023)

Abstract

Latent semantic analysis (LSA) and correspondence analysis (CA) are two techniques that use a singular value decomposition for dimensionality reduction. LSA has been extensively used to obtain low-dimensional representations that capture relationships among documents and terms. In this article, we present a theoretical analysis and comparison of the two techniques in the context of document-term matrices. We show that CA has some attractive properties as compared to LSA, for instance that effects of margins, that is, sums of row elements and column elements, arising from differing document lengths and term frequencies are effectively eliminated so that the CA solution is optimally suited to focus on relationships among documents and terms. A unifying framework is proposed that includes both CA and LSA as special cases. We empirically compare CA to various LSA-based methods on text categorization in English and authorship attribution on historical Dutch texts and find that CA performs significantly better. We also apply CA to a long-standing question regarding the authorship of the Dutch national anthem *Wilhelmus* and provide further support that it can be attributed to the author Datheen, among several contenders.

Keywords: Text data mining; Text classification; Authorship attribution; Information retrieval; Statistical methods; Singular value decomposition

1. Introduction

Latent semantic analysis (LSA) is a well-known method used in computational linguistics that uses singular value decomposition (SVD) for dimensionality reduction in order to extract contextual and usage-based representations of words from textual corpora (Landauer and Dumais, 1997; Jiao and Zhang, 2021). We focus here on LSA of document-term matrices; the rows of the document-term matrix correspond to the documents and the columns to the terms, and the elements are frequencies, that is, the number of occurrences of each term in each document. Documents may have different lengths and margins of documents refer to the marginal frequencies of documents, namely the sum of each row of the document-term matrix; also, terms may be more or less often used and margins of terms refer to the marginal frequencies of terms, namely the sum of each column of the document-term matrix.

Among many other tasks (Di Gangi, Bosco, and Pilato, 2019; Tseng *et al*., 2019; Phillips *et al*., 2021; Hassani, Iranmanesh, and Mansouri, 2021; Ren and Coutanche, 2021; Gupta and Patel, 2021; Kalmukov, 2022), LSA has been used extensively for information retrieval (Zhang, Yoshida, and Tang, 2011; Patil, 2022), by using associations between documents and terms

(Dumais *et al.*, 1988; Deerwester *et al.*, 1990; Dumais, 1991). The exact factorization achieved via SVD has been shown to achieve solutions comparable in some ways to those obtained by modern neural network-based techniques (Levy and Goldberg, 2014; Levy, Goldberg, and Dagan, 2015), commonly used to obtain dense word representations from textual corpora (Jurafsky and Martin, 2021).

Correspondence analysis (CA) is a popular method for the analysis of contingency tables (Greenacre, 1984, Hou and Huang, 2020; Van Dam *et al.*, 2021; Van Dam *et al.*, 2021). It provides a graphical display of dependence between rows and columns of a two-way contingency table (Greenacre and Hastie, 1987). Like LSA, CA is a dimensionality reduction method. The methods have much in common as both use SVD. In both cases, after dimensionality reduction, many text mining tasks, such as text clustering, may be performed in the reduced dimensional space rather than in the higher-dimensional space provided by the raw document-term matrix.

While a few empirical comparisons of LSA and CA, with mixed results, can be found in the literature, a comprehensive theoretical comparison is lacking. For example, Morin (1999) compared the two methods in the automatic exploration of themes in texts. Séguéla and Saporta (2011) compared the performance of CA and LSA with several weighting functions in a document clustering task and found that CA gave better results. On the other hand, Séguéla and Saporta (2013) compared the performance of CA and LSA with TF-IDF on a recommender system but found that CA performs less well.

The present article presents a theoretical comparison of the two techniques and places them in a unifying framework. We show that CA has some favorable properties over LSA, such as a clear interpretation of the distances between documents and between terms of the original matrix, and a clear relation to statistical independence of documents and terms. Also, CA can eliminate the margins of documents and terms simultaneously. Second, we empirically evaluate and compare the two techniques, by applying them to text categorization and authorship attribution in two languages. For text categorization, we use the BBCNews, BBCSport, and 20 Newsgroups datasets in English. In authorship attribution, we evaluate the two techniques on a large set of historical Dutch texts written by six well-known Dutch authors of the sixteenth century. Here, we additionally use CA to determine the unknown authorship of *Wilhelmus*, the national anthem of the Netherlands, whose authorship is controversial: CA attributes *Wilhelmus* to the author Datheen, out of the six contemporary contenders. To the best of our knowledge, this is the first application of CA to the *Wilhelmus*. In both cases, we find that CA performs better.

The rest of the article is organized as follows. Sections 2 and 3 elaborate on the techniques LSA and CA in turn. A unifying framework is proposed in Section 4. In Section 5, we compare LSA and CA in text categorization using the BBCNews, BBCSport, and 20 Newsgroups datasets. Section 6 evaluates the performance of LSA and CA for authorship attribution of documents where the author is known, and of the *Wilhelmus*, whose author is unknown. The article ends with a conclusion.

2. Latent semantic analysis

LSA has been extensively used for improving information retrieval by using the associations between documents and terms (Dumais *et al.*, 1988; Deerwester *et al.*, 1990), among many other tasks. Since individual terms provide incomplete and unreliable evidence about the meaning of a document, in part due to synonymy and polysemy, individual terms are replaced with derived underlying (latent) semantic factors. Although LSA is a very well-known technique, we first present a detailed analysis of the mathematics involved in LSA here as this is usually not found in the literature, and in a later section, it will help in making the comparison between LSA and CA explicit. We start with LSA of the raw document-term matrix and then discuss LSA of weighted matrices. The weighted matrices we study here include (i) a matrix with row-normalized elements with L1, that is, for each row the elements are divided by the row sum (the L1 norm), so that the

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Table 1. A document-term matrix $\boldsymbol{F}$: size 6×6

	lion	tiger	cheetah	jaguar	porsche	ferrari
doc1	2	2	1	2	0	0
doc2	2	3	3	3	0	0
doc3	1	1	1	1	0	0
doc4	2	2	2	3	1	1
doc5	0	0	0	1	1	1
doc6	0	0	0	2	1	2

sum of the elements of each row is 1; (ii) a matrix with row-normalized elements with L2, that is, for each row the elements are divided by the square root of sum of squares of these elements (the L2 norm), so that the sum of squares of the elements of each row is 1; and (iii) a matrix that is transformed by TF-IDF.

The discussion is illustrated using a toy dataset, with the aim to present a clear view of the properties of the dataset captured by LSA and CA; see Table 1. The toy dataset has six rows, the documents, and six columns, the terms, with the frequency of occurrence of terms in each document in the cells (Aggarwal, 2018). Based on term frequencies in each document, the first three documents can be considered to primarily refer to *cats*, the last two primarily to *cars*, and the fourth document to both. The fourth term, *jaguar*, is polysemous because it can refer to either a cat or a car. We will see below how the LSA approaches, and later CA, represent these properties in the data.

2.1 LSA of raw document-term matrix

LSA is an application of the mathematical tool SVD, and can take many forms, depending on the matrix analyzed. We start our discussion of LSA with the SVD of a raw document-term matrix $\boldsymbol{F}$, having size $m \times n$, with elements f_{ij}, $i = 1, \ldots, m$ and $j = 1, \ldots, n$ (Berry, Dumais, and O'Brien, 1995; Deisenroth, Faisal, and Ong, 2020). Without loss of generality we assume that $n \geq m$ and $\boldsymbol{F}$ has full rank.

SVD can be used to decompose $\boldsymbol{F}$ into a product of three matrices: $\boldsymbol{U}^f$, $\boldsymbol{\Sigma}^f$, and $\boldsymbol{V}^f$, namely

$$\boldsymbol{F} = \boldsymbol{U}^f \boldsymbol{\Sigma}^f (\boldsymbol{V}^f)^T \tag{1}$$

Here, $\boldsymbol{U}^f$ is a $m \times m$ matrix with orthonormal columns called left singular vectors so that $(\boldsymbol{U}^f)^T \boldsymbol{U}^f = \boldsymbol{I}$, $\boldsymbol{V}^f$ is a $n \times m$ matrix with orthonormal columns called right singular vectors so that $(\boldsymbol{V}^f)^T \boldsymbol{V}^f = \boldsymbol{I}$, and $\boldsymbol{\Sigma}^f$ is a $m \times m$ diagonal matrix with singular values on the diagonal in descending order.

We denote the first k columns of $\boldsymbol{U}^f$ as the $m \times k$ matrix $\boldsymbol{U}^f_k$, the first k columns of $\boldsymbol{V}^f$ as the $n \times k$ matrix $\boldsymbol{V}^f_k$, and the k largest singular values on the diagonal of $\boldsymbol{\Sigma}^f$ as the $k \times k$ matrix $\boldsymbol{\Sigma}^f_k$ ($k \leq m$). Then $\boldsymbol{U}^f_k \boldsymbol{\Sigma}^f_k (\boldsymbol{V}^f_k)^T$ provides the optimal rank-k approximation of $\boldsymbol{F}$ in a least-squares sense. That is, $\boldsymbol{X} = \boldsymbol{U}^f_k \boldsymbol{\Sigma}^f_k (\boldsymbol{V}^f_k)^T$ minimizes Equation (2) among all matrices $\boldsymbol{X}$ of rank k:

$$||\boldsymbol{F} - \boldsymbol{X}||^2_F = \sum_i \sum_j (f_{ij} - x_{ij})^2 \tag{2}$$

The idea is that the matrix $\boldsymbol{U}^f_k \boldsymbol{\Sigma}^f_k (\boldsymbol{V}^f_k)^T$ captures the major associational structure in the matrix and throws out noise (Dumais *et al.*, 1988; Dumais, 1991). The total sum of squared singular values

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

is equal to $\text{tr}((\boldsymbol{\Sigma}^f)^2)$, where tr is the sum of elements on the main diagonal of a square matrix. The proportion of the total sum of squared singular values explained by the rank k approximation is $\text{tr}((\boldsymbol{\Sigma}_k^f)^2)/\text{tr}((\boldsymbol{\Sigma}^f)^2)$.

SVD can also be interpreted geometrically. As $\boldsymbol{F}$ is of size $m \times n$, each row of $\boldsymbol{F}$ can be represented as a point in an n-dimensional space with the row elements as coordinates, and each column can be represented as a point in an m-dimensional space with the column elements as coordinates. In a rank-k approximation, where $k < (m, n)$, each of the original m documents and n terms is approximated by only k coordinates. Thus, SVD projects the sum of squared Euclidean distances from these row (column) points to the origin in the n (m)-dimensional space as much as possible to a lower, a k-dimensional space. The Euclidean distances between the rows of $\boldsymbol{F}$ are approximated by the Euclidean distances between the rows of $\boldsymbol{U}_k^f \boldsymbol{\Sigma}_k^f$ from below, and the Euclidean distances between the rows of $\boldsymbol{F}^T$ are approximated by the Euclidean distances between the rows of $\boldsymbol{V}_k^f \boldsymbol{\Sigma}_k^f$ from below.

The choice of k is crucial in many applications (Albright, 2004). A lower rank approximation cannot always express prominent relationships in text, whereas the higher rank approximation may add useless noise. How to choose k is an open issue (Deerwester *et al.*, 1990). In practice, the value of k is selected such that a certain criterion is satisfied, for example, the proportion of explained total sum of squared singular values is at least a prespecified proportion. Also, the use of a scree plot, showing the decline in subsequent squared singular values, can be considered.

As $\boldsymbol{F}$ is a non-negative matrix, the first column vectors in $\boldsymbol{U}$ and $\boldsymbol{V}$ have the special property that the elements of the vectors depart in the same direction from the origin (Perron, 1907; Frobenius, 1912; Hu *et al.*, 2003). We give an intuitive geometric explanation for the m rows of $\boldsymbol{F}$. Each row is a vector in the non-negative n-dimensional subspace of R^n. As a result, the first singular vector, being in the middle of the m vectors, is also in the non-negative n-dimensional subspace of R^n. As each vector is the non-negative subspace, the angle between each vector with the first singular vector is between 0° and 90°, and therefore the projection of each of the m vectors on the first singular vector, corresponding to the elements of $\boldsymbol{U}_1$, is non-negative (or each is non-positive, as we will discuss now). The same holds for the columns of $\boldsymbol{F}$ and the first singular vector $\boldsymbol{V}_1$. The reason that the elements of $\boldsymbol{U}_1$ and $\boldsymbol{V}_1$ are all either non-negative or non-positive is that $\boldsymbol{U}_1^f \boldsymbol{\Sigma}_1^f (\boldsymbol{V}_1^f)^T = -\boldsymbol{U}_1^f \boldsymbol{\Sigma}_1^f (-\boldsymbol{V}_1^f)^T$, as the singular values are defined to be non-negative. As the lengths of the row vectors in n-dimensional space to the origin are influenced by the sizes of the documents (i.e., the marginal frequencies), larger documents have larger projections on the first singular vector, and the first dimension mainly displays differences in the sizes of the margins.

As it turns out, the raw document-term matrix $\boldsymbol{F}$ in Table 1 does not have full rank; its rank is 5. The SVD of $\boldsymbol{F}$ in Table 1 is

$$\boldsymbol{F} = \boldsymbol{U}^f \boldsymbol{\Sigma}^f (\boldsymbol{V}^f)^T$$

$$= \begin{bmatrix} -0.411 & 0.175 & 0.825 & 0.252 & -0.239 \\ -0.646 & 0.314 & -0.562 & 0.301 & -0.279 \\ -0.232 & 0.127 & 0.034 & -0.099 & 0.503 \\ -0.562 & -0.203 & 0.044 & -0.603 & 0.333 \\ -0.099 & -0.456 & -0.024 & -0.404 & -0.672 \\ -0.186 & -0.778 & -0.034 & 0.556 & 0.223 \end{bmatrix} \begin{bmatrix} 8.425 & 0 & 0 & 0 & 0 \\ 0 & 3.261 & 0 & 0 & 0 \\ 0 & 0 & 0.988 & 0 & 0 \\ 0 & 0 & 0 & 0.574 & 0 \\ 0 & 0 & 0 & 0 & 0.272 \end{bmatrix}$$

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Table 2. The singular values, the squares of singular values, and the proportion of explained total sum of squared singular values (PSSSV) for each dimension of LSA of $\boldsymbol{F}$, of $\boldsymbol{F}^{L1}$, of $\boldsymbol{F}^{L2}$, and of $\boldsymbol{F}^{\text{TF-IDF}}$

Methods	Items	dim1	dim2	dim3	dim4	dim5
LSA-RAW	Singular value	8.425	3.261	0.988	0.574	0.272
	Square of singular value	70.985	10.635	0.976	0.330	0.074
	PSSSV	0.855	0.128	0.012	0.004	0.001
LSA-NROWL1	Singular value	1.070	0.692	0.123	0.114	0.046
	Square of singular value	1.146	0.479	0.015	0.013	0.002
	PSSSV	0.692	0.289	0.009	0.008	0.001
LSA-NROWL2	Singular value	2.095	1.228	0.239	0.198	0.092
	Square of singular value	4.388	1.507	0.057	0.039	0.009
	PSSSV	0.731	0.251	0.009	0.007	0.001
LSA-TFIDF	Singular value	11.878	5.898	1.565	1.017	0.449
	Square of singular value	141.088	34.782	2.451	1.034	0.202
	PSSSV	0.786	0.194	0.014	0.006	0.001

$$\begin{bmatrix} -0.412 & 0.214 & 0.655 & -0.344 & 0.486 \\ -0.488 & 0.311 & 0.087 & 0.180 & -0.540 \\ -0.440 & 0.257 & -0.748 & -0.259 & 0.339 \\ -0.611 & -0.369 & 0.039 & 0.366 & -0.148 \\ -0.101 & -0.441 & -0.014 & -0.783 & -0.426 \\ -0.123 & -0.679 & -0.048 & 0.186 & 0.392 \end{bmatrix}^T \tag{3}$$

For the raw matrix, LSA-RAW in Table 2 shows the singular values, the squares of the singular values, and the proportions of explained total sum of squared singular values (denoted as PSSSV). Together, the first two dimensions account for $0.855 + 0.128 = 0.983$ of the total sum of squared singular values. Therefore, the documents and the terms can be approximated adequately in a two-dimensional representation using $\boldsymbol{U}_2^f\boldsymbol{\Sigma}_2^f$ and $\boldsymbol{V}_2^f\boldsymbol{\Sigma}_2^f$ as coordinates. As the Euclidean distances between the documents and between the terms in the two-dimensional representation, that is, between the rows of $\boldsymbol{U}_2^f\boldsymbol{\Sigma}_2^f$ and the rows of $\boldsymbol{V}_2^f\boldsymbol{\Sigma}_2^f$, approximate the Euclidean distances between rows and between columns of the original matrix $\boldsymbol{F}$, such a two-dimensional representation simplifies the interpretation of the matrix considerably.

On the other hand, it is somewhat more difficult to examine the relation between a document and a term. The reason is that, by choosing a Euclidean distance representation both for the documents and for terms, the singular values are used *twice* in the coordinates $\boldsymbol{U}_2^f\boldsymbol{\Sigma}_2^f$ and $\boldsymbol{V}_2^f\boldsymbol{\Sigma}_2^f$, and the inner product of coordinates of a document and coordinates of a term does not approximate the corresponding value in $\boldsymbol{F}$. Directions from the origin can be interpreted, though, as the double use of the singular values only leads to relatively reduced coordinates on the second dimension in comparison to the coordinates on the first dimension.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

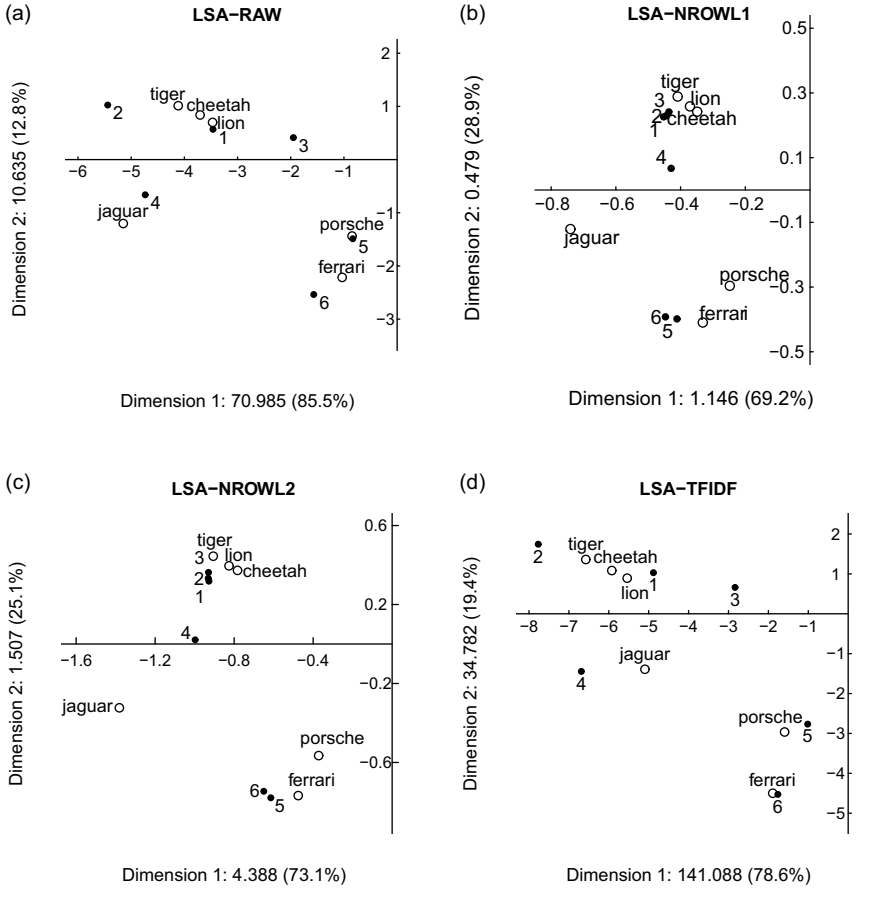

Figure 1. A two-dimensional plot of documents and terms (a) for raw matrix $\boldsymbol{F}$; (b) for row-normalized data $\boldsymbol{F}^{L1}$; (c) for row-normalized data $\boldsymbol{F}^{L2}$; and (d) for matrix $\boldsymbol{F}^{\text{TF-IDF}}$.

The two-dimensional representation of LSA-RAW is shown in Figure 1a. In Figure 1a, Euclidean distances between documents and between terms reveal the similarity of documents and terms, respectively. For example, documents 5 and 6 are close, and similar in the sense that their Euclidean distance is small. For these two documents, the Euclidean distance in the matrix $\boldsymbol{F}$ is 1.414, and in the first two dimensions it is 1.279, so the first two dimensions provide an adequate representation of their similarity. The value 1.279 is much smaller than the Euclidean distances between documents 5 and 1 (3.338), 5 and 2 (5.248), 5 and 3 (2.205), and 5 and 4 (3.988) as well as the Euclidean distances between documents 6 and 1 (3.638), 6 and 2 (5.262), 6 and 3 (2.975), and 6 and 4 (3.681). On the first dimension, all documents and terms have a negative coordinate (see above). There is an order of 5, 6, 3, 1, 4, and 2 on the first dimension. This order is related to the row margins of Table 1, where 2 and 4 have the highest frequencies and therefore are further away from the origin. Overall, the two-dimensional representation of the documents reveals a mix of the sizes of the documents, the row margins $\Sigma_j f_{ij}$, and the relative use of the terms by the documents, that is, for row i this is the vector of elements $f_{ij}/\Sigma_j f_{ij}$, also known as the *row profile* for row i. This mix makes the graphic representation difficult to interpret. Similarly, *porsche* and *ferrari* are lower left but close to the origin, *tiger*, *cheetah*, and *lion* are upper left and further away from the origin, and *jaguar* is far away at the lower left. Also there is a mix of the sizes of the terms, that is, for column j this is column margin $\Sigma_i f_{ij}$, and the relative use of the documents by the terms,

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

that is, for column j this is the vector of elements $f_{ij}/\Sigma_i f_{ij}$, also known as the *column profile* for column j. The terms *porsche* and *ferrari* are related to documents 5 and 6 as they have the same position w.r.t. the origin, and similarly for *tiger*, *cheetah*, and *lion* to documents 1, 2, and 3, and *jaguar* to document 4.

Although the first dimension accounts for 85.5% of the total sum of squared singular values, it provides little information about the relations among documents and terms. In particular, from Table 1 we expect that documents 1–3 are similar, documents 5 and 6 are similar, and document 4 is in-between; term *jaguar* is between cat terms (*tiger*, *cheetah*, and *lion*) and car terms (*porsche* and *ferrari*), but we cannot see that from the first dimension. This is because the margins of Table 1 play a dominant role in the first dimension.

2.2 LSA of weighted document-term matrix

Weighting can be used to prevent differential lengths of documents from having differential effects on the representation, or be used to impose certain preconceptions of which terms are more important (Deerwester *et al.*, 1990). The frequencies f_{ij} in the raw document-term matrix $\boldsymbol{F}$ can be transformed with the aim to provide a better approximation of the interrelations between documents and terms (Nakov, Popova, and Mateev, 2001). The weight w_{ij} for term j in document i is normally expressed as a product of three components (Salton and Buckley, 1988; Kolda and O'leary, 1998; Ab Samat *et al.*, 2008):

$$w_{ij} = L(i,j) \times G(j) \times N(i) \tag{4}$$

where the local weighting $L(i,j)$ is the weight of term j in document i, the global weighting $G(j)$ is the weight of the term j in the entire document set, and $N(i)$ is the normalization component for document i.

When $L(i,j) = f(i,j)$, $G(j) = 1$, and $N(i) = 1$, the weighted $\boldsymbol{F}$ is equal to $\boldsymbol{F}$. In matrix notation, Equation (4) can be expressed as $\boldsymbol{W} = \boldsymbol{NLG}$, where $\boldsymbol{N}$ is a diagonal matrix with diagonal elements $N(i)$ and $\boldsymbol{G}$ is a diagonal matrix with diagonal elements $G(j)$. Notice that pre- or post-multiplying by a diagonal matrix leaves the rank of the matrix $\boldsymbol{L}$ intact.

We examine two common ways to weight f_{ij}. One is row normalization (Salton and Buckley, 1988; Ab Samat *et al.*, 2008) with L1 and L2. The other is TF-IDF (Dumais, 1991).

2.2.1 SVD of matrix with row-normalized elements with L1

In row-normalized weighting with L1, we use Equation (4) with $L(i,j) = f_{ij}$, $G(j) = 1$, and $N(i) = 1/\sum_{j=1}^{n} f_{ij}$ and apply an SVD to this transformed matrix that we denote as $\boldsymbol{F}^{L1}$, which consists of the row profiles of $\boldsymbol{F}$. See Table 3. The last row, the average row profile, is the row profile of the column margins of Table 1.

We perform LSA of $\boldsymbol{F}^{L1}$ and find Table 2, part LSA-NROWL1. This shows that a rank 2 matrix approximates the data well as $0.692 + 0.289 = 0.981$ of the total sum of squared singular values is explained by these two dimensions. The first two columns of LSA of $\boldsymbol{F}^{L1}$ can be used to approximate $\boldsymbol{F}^{L1}$; see Equation (5):

$$\boldsymbol{F}^{L1} \approx \boldsymbol{U}_2^{L1}\boldsymbol{\Sigma}_2^{L1}(\boldsymbol{V}_2^{L1})^T = \begin{bmatrix} -0.423 & 0.327 \\ -0.415 & 0.332 \\ -0.408 & 0.349 \\ -0.401 & 0.097 \\ -0.384 & -0.575 \\ -0.417 & -0.567 \end{bmatrix} \begin{bmatrix} 1.070 & 0 \\ 0 & 0.692 \end{bmatrix} \begin{bmatrix} -0.347 & 0.374 \\ -0.382 & 0.417 \\ -0.326 & 0.350 \\ -0.692 & -0.174 \\ -0.232 & -0.428 \\ -0.310 & -0.592 \end{bmatrix}^T \tag{5}$$

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Table 3. Row profiles of $\boldsymbol{F}$

	lion	tiger	cheetah	jaguar	porsche	ferrari	Total
doc1	0.286	0.286	0.143	0.286	0.000	0.000	1.000
doc2	0.182	0.273	0.273	0.273	0.000	0.000	1.000
doc3	0.250	0.250	0.250	0.250	0.000	0.000	1.000
doc4	0.182	0.182	0.182	0.273	0.091	0.091	1.000
doc5	0.000	0.000	0.000	0.333	0.333	0.333	1.000
doc6	0.000	0.000	0.000	0.400	0.200	0.400	1.000
Average row profile	0.171	0.195	0.171	0.293	0.073	0.098	1.000

Table 4. A row-normalized document-term matrix $\boldsymbol{F}^{L2}$

	lion	tiger	cheetah	jaguar	porsche	ferrari
doc1	0.555	0.555	0.277	0.555	0.000	0.000
doc2	0.359	0.539	0.539	0.539	0.000	0.000
doc3	0.500	0.500	0.500	0.500	0.000	0.000
doc4	0.417	0.417	0.417	0.626	0.209	0.209
doc5	0.000	0.000	0.000	0.577	0.577	0.577
doc6	0.000	0.000	0.000	0.667	0.333	0.667

Documents and terms can be projected on a two-dimensional space using $\boldsymbol{U}_2^{L1}\boldsymbol{\Sigma}_2^{L1}$ and $\boldsymbol{V}_2^{L1}\boldsymbol{\Sigma}_2^{L1}$ as coordinates; see Figure 1b. In this representation, documents 1, 2, and 3 are quite close, and so are 5 and 6. Also, the terms *ferrari* and *porsche* are close and related to 5 and 6, *tiger*, *lion*, and *cheetah* are close and related to 1, 2, and 3.

Although the first dimension accounts for 69.2% of the total sum of squared singular values, this dimension does not provide information about different use of terms by the documents as all documents have a similar coordinate. This is caused by the same marginal value 1 for each of the documents in $\boldsymbol{F}^{L1}$, which leads to almost the same distance from the origin. Also, we would expect *jaguar* to be in between cat terms (*tiger*, *cheetah*, and *lion*) and car terms (*porsche* and *ferrari*), but on the first dimension it appears as a separate, third group. This is caused by the high values in its column in $\boldsymbol{F}^{L1}$, which lead to a larger distance from the origin.

2.2.2 SVD of matrix with row-normalized elements with L2

In row-normalized weighting with L2, we use Equation (4) with $L(i,j) = f_{ij}$, $G(j) = 1$, and $N(i) = 1/\sqrt{\sum_{j=1}^{n} f_{ij}^2}$. The transformed matrix, denoted as $\boldsymbol{F}^{L2}$, is shown in Table 4. We then perform LSA on Table 4. Table 2, part LSA-NROWL2, indicates that a rank 2 matrix approximates the data well, as the sum of the PSSSV of the first two dimensions $0.731 + 0.251 = 0.982$ contributes to 98.2% of the total sum of squared singular values. The first two columns of LSA of $\boldsymbol{F}^{L2}$ can be used to approximate $\boldsymbol{F}^{L2}$; see Equation (6):

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Table 5. A document-term matrix $\boldsymbol{F}^{\text{TF-IDF}}$

	lion	tiger	cheetah	jaguar	porsche	ferrari
doc1	3.170	3.170	1.585	2	0	0
doc2	3.170	4.755	4.755	3	0	0
doc3	1.585	1.585	1.585	1	0	0
doc4	3.170	3.170	3.170	3	2	2
doc5	0.000	0.000	0.000	1	2	2
doc6	0.000	0.000	0.000	2	2	4

$$\boldsymbol{F}^{L2} \approx \boldsymbol{U}_2^{L2}\boldsymbol{\Sigma}_2^{L2}(\boldsymbol{V}_2^{L2})^T = \begin{bmatrix} -0.443 & 0.259 \\ -0.445 & 0.271 \\ -0.444 & 0.295 \\ -0.476 & 0.017 \\ -0.293 & -0.635 \\ -0.310 & -0.608 \end{bmatrix} \begin{bmatrix} 2.095 & 0 \\ 0 & 1.228 \end{bmatrix} \begin{bmatrix} -0.394 & 0.323 \\ -0.432 & 0.362 \\ -0.374 & 0.304 \\ -0.659 & -0.263 \\ -0.178 & -0.460 \\ -0.227 & -0.625 \end{bmatrix}^T \tag{6}$$

Documents and terms can be projected on a two-dimensional space using $\boldsymbol{U}_2^{L2}\boldsymbol{\Sigma}_2^{L2}$ and $\boldsymbol{V}_2^{L2}\boldsymbol{\Sigma}_2^{L2}$ as coordinates; see Figure 1c. In this representation, documents 1, 2, and 3 are quite close, and so are 5 and 6. Also, the terms *ferrari* and *porsche* are close and related to 5 and 6, *tiger*, *lion*, and *cheetah* are close and related to 1, 2, and 3.

Although the first dimension accounts for 73.1% of the total sum of squared singular values, and so, a major portion of the information in the matrix, we do not find the important aspect in the data that document 4 should be in between documents 1–3 on the one hand and documents 5–6 on the other hand on this dimension. This is caused by the high values in the row for doc4 in Table 4, which lead to a larger distance from the origin than the other documents have. Also, we would expect *jaguar* to be in between cat terms (*tiger*, *cheetah*, and *lion*) and car terms (*porsche* and *ferrari*), but on the first dimension it appears as a separate, third group. This is caused by the high values in its column in Table 4, which lead to a larger distance from the origin.

2.2.3 SVD of the TF-IDF matrix

TF-IDF is one commonly used transformation of text data. We use Equation (4) with $L(i,j) = f_{ij}$, $G(j) = 1 + \log(\frac{n\text{docs}}{df_j})$, and $N(i) = 1$, one form of TF-IDF, where $n$docs is the number of documents in the set and df_j is the number of documents where term j appears, and then apply an SVD to this transformed matrix that we denote as $\boldsymbol{F}^{\text{TF-IDF}}$; see Table 5. As is common in the literature, here we choose 2 as the base of the logarithmic function.

We perform LSA of Table 5 and find Table 2, part LSA-TFIDF. This shows that a rank 2 matrix approximates the data well as $0.786 + 0.194 = 0.980$ of the total sum of squared singular values is explained by these two dimensions. The matrix $\boldsymbol{F}^{\text{TF-IDF}}$ in Table 5 is approximated in the first two dimensions as follows:

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

$$\boldsymbol{F}^{\text{TF-IDF}} \approx \boldsymbol{U}_2^{\text{TF-IDF}} \boldsymbol{\Sigma}_2^{\text{TF-IDF}} (\boldsymbol{V}_2^{\text{TF-IDF}})^T$$

$$= \begin{bmatrix} -0.411 & 0.175 \\ -0.654 & 0.296 \\ -0.239 & 0.112 \\ -0.563 & -0.245 \\ -0.086 & -0.469 \\ -0.148 & -0.768 \end{bmatrix} \begin{bmatrix} 11.878 & 0 \\ 0 & 5.898 \end{bmatrix} \begin{bmatrix} -0.466 & 0.151 \\ -0.554 & 0.231 \\ -0.499 & 0.184 \\ -0.429 & -0.236 \\ -0.134 & -0.502 \\ -0.159 & -0.763 \end{bmatrix}^T \tag{7}$$

Figure 1d is a two-dimensional plot of the documents and terms using $\boldsymbol{U}_2^{\text{TF-IDF}} \boldsymbol{\Sigma}_2^{\text{TF-IDF}}$ and $\boldsymbol{V}_2^{\text{TF-IDF}} \boldsymbol{\Sigma}_2^{\text{TF-IDF}}$ as coordinates for the 6×6 sample document-term matrix $\boldsymbol{F}^{\text{TF-IDF}}$. The configuration of documents in Figure 1d is very similar to that in Figure 1a. The configuration of terms in Figure 1d is different from that of terms in Figure 1a. In Figure 1d, there is an order of *porsche*, *ferrari*, *jaguar*, *lion*, *cheetah*, and *tiger* on the first dimension, whereas in Figure 1a, there is an order of *porsche*, *ferrari*, *lion*, *cheetah*, *tiger*, and *jaguar* on the first dimension. Compared with Figure 1a, the first dimension of Figure 1d shows that *jaguar* is in between cat terms (*tiger*, *cheetah*, and *lion*) and car terms (*porsche* and *ferrari*).

2.2.4 Out-of-sample documents

Representing out-of-sample documents in the k-dimensional subspace of LSA is important for many applications. Suppose an out-of-sample document $\boldsymbol{d}$ is a row vector. To represent $\boldsymbol{d}$ in lower-dimensional space, first the out-of-sample document $\boldsymbol{d}$ can be transformed in the same way as the original documents (Dumais, 1991). Transformations for the above four applications of LSA are $\boldsymbol{d}_w^f = \boldsymbol{d}$, $\boldsymbol{d}_w^{L1} = \boldsymbol{d} / \sum_{j=1}^n d_j$, $\boldsymbol{d}_w^{L2} = \boldsymbol{d} / \sqrt{\sum_{j=1}^n d_j^2}$, and $\boldsymbol{d}_w^{\text{TF-IDF}} = [d_1 G(1), \cdots, d_n G(n)]$. The coordinates of the out-of-sample document $\boldsymbol{d}$ in LSA-RAW, LSA-NROWL1, LSA-NROWL2, and LSA-TFIDF are then calculated by $\boldsymbol{d}_w^f \boldsymbol{V}^f$, $\boldsymbol{d}_w^{L1} \boldsymbol{V}^{L1}$, $\boldsymbol{d}_w^{L2} \boldsymbol{V}^{L2}$, and $\boldsymbol{d}_w^{\text{TF-IDF}} \boldsymbol{V}^{\text{TF-IDF}}$, respectively (Aggarwal, 2018).

2.3 Conclusions regarding LSA of different matrices

In the raw document-term matrix, the relationships among the documents and terms is blurred by differences in margins arising from differing document lengths and marginal term frequencies. Thus LSA of the raw matrix leads to a mix of margins, and relationships among documents and terms. In order to provide a better approximation of the interrelations between documents and terms, weighting schemes were used.

Normalizations of the documents have a beneficial effect. Yet, the properties of the frequencies that are evident from Table 1 where we expect, for example, that *jaguar* lies in between *porsche* and *ferrari* on the one hand and *tiger*, *cheetah*, and *lion* on the other hand, are not fully represented on the first dimension. This is due to the fact that the column margins of Tables 3 and 4 still play a role on the first dimension. The TF-IDF transformation also has a positive effect. Yet LSA is not successful. For example, we expect that documents 1–3 are similar, 5 and 6 are similar, and document 4 is in-between, but this order is not found in the first dimension. This is due to the fact that the row margins of Table 5 still play a role on the first dimension.

Generally, solutions of LSA have the drawback that they include the effect of the margins as well as the dependence. In the first dimension, these margins play a dominant role as all points

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

depart in the same direction from the origin. We can try to repair this property of LSA, by applying transformations of the rows and columns of Table 1 simultaneously. However, the transformations appear ad hoc. Instead, we present in the next section a different technique, which better fits the properties of the data: CA.

3. Correspondence analysis

CA provides a low-dimensional representation of the interaction or dependence between the rows and columns of the contingency table (Greenacre and Hastie, 1987), which can be used to reveal the structure in the data (Hayashi, 1992). CA has been proposed multiple times, apparently independently, emphasizing different properties of the technique (Gifi, 1990). Some important contributions are provided in the Japanese literature, by Hayashi (1956, 1992), who emphasizes the property of CA that it maximizes the correlation coefficient between the row and column variable by assigning numerical scores to these variables; in the French literature, by Benzécri (1973), who emphasizes a distance interpretation, where Greenacre (1984) expressed Benzécri's work in a more convenient mathematical notation; and in the Dutch literature, by Gifi (1990) and Michailidis and De Leeuw (1998), who emphasize optimal scaling properties. We present CA here mainly from the French perspective.

The aim of CA as developed by Benzécri is to find a representation of the rows (columns) of frequency matrix $\boldsymbol{F}$ in such a way that Euclidean distances between the rows (columns) in the representation correspond to so-called χ^2-distances between rows (columns) of $\boldsymbol{F}$ (Gifi, 1990). We work with $\boldsymbol{P}$ with elements $p_{ij} = f_{ij}/f_{++}$, where f_{++} is the sum of all elements of $\boldsymbol{F}$. In the χ^2-distance, profiles play an important role. The squared χ^2-distance between the kth row profile with elements p_{kj}/r_k and the lth row profile with elements p_{lj}/r_l is

$$\delta_{kl}^2 = \sum_j \frac{\left(p_{kj}/r_k - p_{lj}/r_l\right)^2}{c_j} \tag{8}$$

where r_i (also called the average column profile) and c_j (the average row profile) are the row and column sums of $\boldsymbol{P}$, respectively. Thus, the difference between the jth elements of the two profiles is weighted by column margin (i.e., the last row of Table 3), c_j, so that this difference plays a relatively more important role in the χ^2-distance if it stems from a column having a small value c_j.

A representation where Euclidean distances between the rows of the matrix are equal to χ^2-distances is found as follows. In matrix notation, the matrix whose Euclidean distances between the rows are equal to χ^2-distances between rows of $\boldsymbol{F}$ is equal to $\boldsymbol{D}_r^{-1}\boldsymbol{P}\boldsymbol{D}_c^{-\frac{1}{2}}$, where $\boldsymbol{D}_r$ is a diagonal matrix with r_i as diagonal elements and $\boldsymbol{D}_c$ is a diagonal matrix with c_j as diagonal elements. Suppose we take the SVD of

$$\boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{P}\boldsymbol{D}_c^{-\frac{1}{2}} = \boldsymbol{U}^{sp}\boldsymbol{\Sigma}^{sp}(\boldsymbol{V}^{sp})^T \tag{9}$$

Here, $\boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{P}\boldsymbol{D}_c^{-\frac{1}{2}}$ is a matrix with standardized proportions, hence the superscripts sp on the right-hand side of the equation. Then, if we premultiply both sides of Equation (9) with $\boldsymbol{D}_r^{-\frac{1}{2}}$, we get

$$\boldsymbol{D}_r^{-1}\boldsymbol{P}\boldsymbol{D}_c^{-\frac{1}{2}} = \boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{U}^{sp}\boldsymbol{\Sigma}^{sp}(\boldsymbol{V}^{sp})^T \tag{10}$$

Thus, a representation using the rows of $\boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{U}^{sp}\boldsymbol{\Sigma}^{sp}$ as row coordinates leads to Euclidean distances between these row points being equal to χ^2-distances between rows of $\boldsymbol{F}$. Similar to Equation (8), we can also define χ^2-distances between the columns of $\boldsymbol{F}$, and in matrix notation this leads to the matrix $\boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{P}\boldsymbol{D}_c^{-1}$. Then, in a similar way as for the χ^2-distances for

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

the rows, Equation (9) can be used as an intermediate step to go to a solution for the columns. Postmultiplying the left- and right-hand sides in Equation (9) by $\boldsymbol{D}_c^{-\frac{1}{2}}$ provides us with the coordinates for a representation where Euclidean distances between the column points (the rows of $\boldsymbol{D}_c^{-\frac{1}{2}}\boldsymbol{V}^{sp}\boldsymbol{\Sigma}^{sp}$ as coordinates for these columns) are equal to χ^2-distances between the columns of $\boldsymbol{F}$. Notice that Equation (9) plays the dual role of an intermediate step in going to a solution both for the rows and the columns.

The matrices $\boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{U}^{sp}\boldsymbol{\Sigma}^{sp}$ and $\boldsymbol{D}_c^{-\frac{1}{2}}\boldsymbol{V}^{sp}\boldsymbol{\Sigma}^{sp}$ have a first column being equal to 1, a so-called artificial dimension. This artificial dimension reflects the fact that the row margins of the matrix $\boldsymbol{D}_r^{-1}\boldsymbol{P}$ with the row profiles of Table 1 are 1 and the column margins of the matrix $\boldsymbol{P}\boldsymbol{D}_c^{-1}$ with the column profiles of Table 1 are 1. This artificial dimension is eliminated by not taking the SVD of $\boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{P}\boldsymbol{D}_c^{-\frac{1}{2}}$ but of $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$, where the elements of $\boldsymbol{E}$ are defined as the product of the margins r_i and c_j. Due to subtracting $\boldsymbol{E}$ from $\boldsymbol{P}$, the rank of $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ is $m-1$, which is 1 less than the rank of $\boldsymbol{F}$. Notice that the elements of $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ are standardized residuals under the independence model, and the sum of squares of these elements yields the so-called total inertia, which is equal to the Pearson χ^2 statistic divided by sample size f_{++}. By taking the SVD of the matrix of standardized residuals, we get

$$\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}} = \boldsymbol{U}^{sr}\boldsymbol{\Sigma}^{sr}(\boldsymbol{V}^{sr})^T \tag{11}$$

and

$$\boldsymbol{D}_r^{-1}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-1} = \boldsymbol{\Phi}^{sr}\boldsymbol{\Sigma}^{sr}(\boldsymbol{\Gamma}^{sr})^T \tag{12}$$

where $\boldsymbol{\Phi}^{sr} = \boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{U}^{sr}$ and $\boldsymbol{\Gamma}^{sr} = \boldsymbol{D}_c^{-\frac{1}{2}}\boldsymbol{V}^{sr}$. We use the abbreviation *sr* for the matrices on the right-hand side of Equation (11) to refer to the matrix of standardized residuals on the left-hand side of the equation. CA simultaneously provides a geometric representation of row profiles and column profiles of Table 1, where the effects of row margins and column margins of Table 1 are eliminated. $\boldsymbol{\Phi}^{sr}$ and $\boldsymbol{\Gamma}^{sr}$ are called standard coordinates of rows and columns, respectively. They have the property that their weighted average is 0 and weighted sum of squares is 1:

$$\mathbf{1}^T\boldsymbol{D}_r\boldsymbol{\Phi}^{sr} = \mathbf{0}^T = \mathbf{1}^T\boldsymbol{D}_c\boldsymbol{\Gamma}^{sr} \tag{13}$$

and

$$(\boldsymbol{\Phi}^{sr})^T\boldsymbol{D}_r\boldsymbol{\Phi}^{sr} = \mathbf{I} = (\boldsymbol{\Gamma}^{sr})^T\boldsymbol{D}_c\boldsymbol{\Gamma}^{sr} \tag{14}$$

Equation (13) reflects the fact that the row and column margins of $\boldsymbol{P}-\boldsymbol{E}$ vanish (Van der Heijden, De Falguerolles, and De Leeuw, 1989).

We can make graphic displays using $\boldsymbol{\Phi}_k^{sr}\boldsymbol{\Sigma}_k^{sr}$ and $\boldsymbol{\Gamma}_k^{sr}\boldsymbol{\Sigma}_k^{sr}$ as coordinates, which has the advantage that Euclidean distances between the points approximate χ^2-distances both for the rows of $\boldsymbol{F}$ and for the columns of $\boldsymbol{F}$, but it has the drawback that $\boldsymbol{\Sigma}_k^{sr}$ is used twice. We can also make graphic displays using $\boldsymbol{\Phi}_k^{sr}\boldsymbol{\Sigma}_k^{sr}$ and $\boldsymbol{\Gamma}_k^{sr}$, or $\boldsymbol{\Phi}_k^{sr}$ and $\boldsymbol{\Gamma}_k^{sr}\boldsymbol{\Sigma}_k^{sr}$. Thus, from Equation (12), this has the advantage that the inner product of the coordinates of a document and the coordinates of a term approximates the corresponding value in $\boldsymbol{D}_r^{-1}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-1}$.

If we choose $\boldsymbol{\Phi}^{sr}\boldsymbol{\Sigma}^{sr}$ for the row points and $\boldsymbol{\Gamma}^{sr}$ for the column points, then CA has the property that the row points are in weighted average of the column points, where the weights are the row profile values. Actually, $\boldsymbol{\Gamma}^{sr}$ can be seen as coordinates for the extreme row profiles projected onto the subspace. The extreme row profiles are totally concentrated into one of the terms. For example, [0, 0, 1, 0, 0, 0] represents the row profile of a document that is totally concentrated into *cheetah*. At the same time, if we choose $\boldsymbol{\Phi}^{sr}$ for the row points and $\boldsymbol{\Gamma}^{sr}\boldsymbol{\Sigma}^{sr}$ for the column points, column points are in weighted average of row points, where the weights are the column profile values. In

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Table 6. The matrix $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ of standardized residuals

	lion	tiger	cheetah	jaguar	porsche	ferrari
doc1	0.115	0.085	−0.028	−0.005	−0.112	−0.129
doc2	0.014	0.091	0.128	−0.019	−0.140	−0.162
doc3	0.060	0.039	0.060	−0.025	−0.084	−0.098
doc4	0.014	−0.016	0.014	−0.019	0.034	−0.011
doc5	−0.112	−0.119	−0.112	0.020	0.260	0.204
doc6	−0.144	−0.154	−0.144	0.069	0.164	0.338

a similar way as for the rows, $\boldsymbol{\Phi}^{sr}$ provide coordinates for the extreme column profiles projected onto the subspace. The relationship between these row points and column points can be shown by rewriting Equation (11) and using Equation (13) as:

$$\boldsymbol{D}_r^{-1}\boldsymbol{P}\boldsymbol{\Gamma}^{sr} = \boldsymbol{\Phi}^{sr}\boldsymbol{\Sigma}^{sr} \tag{15}$$

and

$$\boldsymbol{D}_c^{-1}\boldsymbol{P}^T\boldsymbol{\Phi}^{sr} = \boldsymbol{\Gamma}^{sr}\boldsymbol{\Sigma}^{sr} \tag{16}$$

These equations are called the transition formulas. In fact, using transition formulas is one of the ways in which the solution of CA can be obtained: starting from arbitrary values for the columns, one first centers and standardizes the column coordinates so that the weighted sum is 0 and the weighted sums of squares is 1, next places the rows in the weighted average of the columns, then places the columns in the weighted average of the rows, and so on, until convergence. This is known as reciprocal averaging (Hill, 1973, 1974). Using the transition formula (15), the coordinates of the out-of-sample document $\boldsymbol{d}$ is $(\boldsymbol{d}/\sum_{j=1}^{n} d_j)\boldsymbol{\Gamma}^{sr}$ (Greenacre, 2017).

The origin in the graphic representation for the rows stands for the average row profile, which can be seen as follows. Let $\boldsymbol{D}_r^{-1}\boldsymbol{P}\boldsymbol{D}_c^{-\frac{1}{2}}$ be the matrix where Euclidean distances between the rows are χ^2-distances between rows of $\boldsymbol{F}$. Assume we plot the rows of this matrix using the n elements of each row as coordinates. Then, eliminating the artificial dimension in $\boldsymbol{D}_r^{-1}\boldsymbol{P}\boldsymbol{D}_c^{-\frac{1}{2}}$ leads to the subtraction of the average row profile from each row, as $\boldsymbol{D}_r^{-1}\boldsymbol{E}$ is a matrix with the average row profile in each row. In other words, the cloud of row points is translated to the origin, with the average row profile being exactly in the origin (compare Equation (13): $\mathbf{0}^T = \mathbf{1}^T\boldsymbol{D}_c\boldsymbol{\Gamma}^{sr}$). When two row points are departing in the same way from the origin, they depart in the same way from the average profile, and when two row points are on opposite sides of the origin, they depart in opposite ways from the average profile. If the documents and terms are statistically independent, then $p_{ij}/r_i = c_j$, and all document profiles would lie in the origin. Thus, comparing row profiles with the origin is a way to study the departure from independence and to study the relations between documents and terms. Similarly, the origin in the graphic representation for the columns stands for average column profile.

We now analyze the example discussed in the LSA section. There are three steps to obtain the CA solution. Step 1: make the matrix $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ of standardized residuals; Step 2: compute the SVD of the matrix; Step 3: derive $\boldsymbol{\Phi}^{sr} = \boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{U}^{sr}$ and $\boldsymbol{\Gamma}^{sr} = \boldsymbol{D}_c^{-\frac{1}{2}}\boldsymbol{V}^{sr}$, and postmultiply $\boldsymbol{\Phi}^{sr}$ and $\boldsymbol{\Gamma}^{sr}$ by $\boldsymbol{\Sigma}^{sr}$ to obtain the coordinates. Table 6 shows the matrix $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ of standardized residuals (in lower-case notation, the elements of the matrix are $(p_{ij}-e_{ij})/\sqrt{e_{ij}}$).

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Table 7. The singular values, the inertia, and the proportions of explained total inertia for each dimension of CA

	dim1	dim2	dim3	dim4
Singular value	0.689	0.131	0.124	0.044
Inertia	0.475	0.017	0.015	0.002
The proportion of inertia	0.932	0.034	0.030	0.004

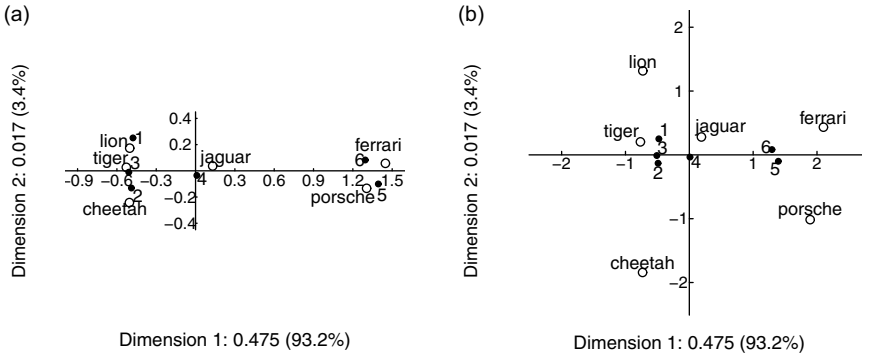

Figure 2. The data of Table 1 using CA for (a) symmetric map and (b) asymmetric map.

We perform an SVD of $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ in Table 6 and find Table 7. Due to subtracting $\boldsymbol{E}$ from $\boldsymbol{P}$, the rank of the matrix in Table 6 is 4, which is 1 less than that in Table 1. The proportion of the total inertia explained by only the first dimension accounts for 0.932 of the total inertia. The matrix $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ in Table 6 is approximated in the first two dimensions as follows:

$$\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}} \approx \boldsymbol{U}_2^{sr}\boldsymbol{\Sigma}_2^{sr}(\boldsymbol{V}_2^{sr})^T$$

$$= \begin{bmatrix} -0.286 & 0.789 \\ -0.368 & -0.517 \\ -0.231 & -0.025 \\ 0.007 & -0.138 \\ 0.547 & -0.206 \\ 0.656 & 0.220 \end{bmatrix} \begin{bmatrix} 0.689 & 0 \\ 0 & 0.131 \end{bmatrix} \begin{bmatrix} -0.301 & 0.544 \\ -0.338 & 0.090 \\ -0.303 & -0.761 \\ 0.102 & 0.152 \\ 0.512 & -0.275 \\ 0.656 & 0.136 \end{bmatrix}^T \tag{17}$$

Figure 2a is the map with a symmetric role for the rows and the columns, having $\boldsymbol{\Phi}_2^{sr}\boldsymbol{\Sigma}_2^{sr}$ and $\boldsymbol{\Gamma}_2^{sr}\boldsymbol{\Sigma}_2^{sr}$ as coordinates. The larger the deviations from document (term) points to the origin are, the larger the dependence between documents and terms. Looking only at the first dimension and document profiles' positions, we can see that the groups furthest apart are documents 1–3 on the left-hand side, opposed to documents 5–6 on the right-hand side. They differ in opposite ways from the average row profile that lies in the origin. For the term points on the first dimension, the cat terms (*tiger*, *cheetah*, and *lion*) lie on the left, and car terms (*porsche* and *ferrari*) on the right.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

They differ in opposite ways from the average column profile. Importantly, CA clearly displays the properties we see in the data matrix, as document 4 lies between documents 1–3 and documents 5–6, and the term *jaguar* lies between cat terms and car terms, unlike all four of the LSA-based analyses presented in Figure 1.

Figure 2b is the asymmetric map with documents in the weighted average of the terms ($\boldsymbol{\Phi}_2^{sr}\boldsymbol{\Sigma}_2^{sr}$ and $\boldsymbol{\Gamma}_2^{sr}$ as coordinates, notice that the position of the documents is identical as in Figure 2a). From this graphic display, we can study the position of the documents as they are in the weighted average of the terms, using the row profile elements as weights. For example, document 1 is closer to *lion* and *tiger* than to *porsche* and *ferrari*, because it has higher profile values than average values on terms *lion* and *tiger* (both 0.286 in comparison with the average profile values 0.171 and 0.195) and lower profile values on the terms *porsche* and *ferrari* (both 0.000 in comparison to 0.073 and 0.098); see Table 3. Thus, document 1 is pulled into the direction of *lion* and *tiger*.

3.1 Conclusions regarding CA

In CA, an SVD is applied to the matrix $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ of standardized residuals. Due to $\boldsymbol{E}$, in CA the effect of the margins is eliminated and a solution only displays the relationships among documents and terms. In CA, all points are scattered around the origin and the origin represents the profile of the row and column margins of $\boldsymbol{F}$.

In comparison, LSA also tries to capture the relationships among documents and terms, which is not easy. The reason is that these relations are blurred by the effect of the margins that are also displayed in the LSA solution. CA does not have this property. Therefore, it appears that CA is a better tool for information retrieval, natural language processing, and text mining.

4. A unifying framework

Here, we present a unifying framework that integrates LSA and CA. This section also serves the purpose of showing their similarities and their differences.

To first summarize LSA (see Section 2.2 for details), a matrix is weighted, and the weighted matrix is decomposed. Assume we start off with the document-term matrix $\boldsymbol{F}$, the row weights of $\boldsymbol{F}$ are collected in the diagonal matrix $\boldsymbol{N}$, the column weights in the diagonal matrix $\boldsymbol{G}$, and there may be local weighting of the elements f_{ij} of $\boldsymbol{F}$ leading to a locally weighted matrix $\boldsymbol{L}$. Thus, the weighted matrix $\boldsymbol{W}$ can be written as the matrix product:

$$\boldsymbol{W} = \boldsymbol{NLG} \tag{18}$$

Subsequently, in LSA, the matrix $\boldsymbol{W}$ is decomposed using SVD into a product of three matrices: the orthonormal matrix $\boldsymbol{U}$, the diagonal matrix $\boldsymbol{\Sigma}$ with singular values in descending order, and the orthonormal matrix $\boldsymbol{V}$, namely

$$\boldsymbol{W} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{19}$$

with

$$\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I} = \boldsymbol{V}^T\boldsymbol{V} \tag{20}$$

Graphic representations are usually made using $\boldsymbol{U}\boldsymbol{\Sigma}$ as coordinates for the rows and $\boldsymbol{V}\boldsymbol{\Sigma}$ for the columns.

In contrast, in CA we take the SVD of the matrix of standardized residuals. Let $\boldsymbol{P}$ be the matrix with proportions $p_{ij} = f_{ij}/f_{++}$, where f_{++} is the sum of all elements of $\boldsymbol{F}$; let $\boldsymbol{E}$ be the matrix with expected proportions under independence $e_{ij} = r_i c_j$, where r_i and c_j are the row and column sums of $\boldsymbol{P}$, respectively; let $\boldsymbol{D_r}$ and $\boldsymbol{D_c}$ be diagonal matrices with row and column sums r_i and c_j, respectively. Thus, the matrix of standardized residuals is $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$. If we take the SVD of this

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

matrix, we get (11),

$$\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{21}$$

In CA, the matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ are further adjusted by:

$$\boldsymbol{\Phi} = \boldsymbol{D}_r^{-\frac{1}{2}}\boldsymbol{U}, \boldsymbol{\Gamma} = \boldsymbol{D}_c^{-\frac{1}{2}}\boldsymbol{V} \tag{22}$$

so that we can write

$$\boldsymbol{D}_r^{-1}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-1} = \boldsymbol{\Phi}\boldsymbol{\Sigma}\boldsymbol{\Gamma}^T \tag{23}$$

with

$$\boldsymbol{\Phi}^T\boldsymbol{D}_r\boldsymbol{\Phi} = \boldsymbol{I} = \boldsymbol{\Gamma}^T\boldsymbol{D}_c\boldsymbol{\Gamma} \tag{24}$$

Graphic representations are usually made using $\boldsymbol{\Phi}\boldsymbol{\Sigma}$ and $\boldsymbol{\Gamma}\boldsymbol{\Sigma}$ as coordinates for the rows and columns, respectively.

This brings us to the point where we can formulate a unifying framework. We distinguish the matrix to be analyzed and the decomposition of this matrix. For the matrix to be analyzed, the weighted matrix defined in (18) can be used by LSA as well as by CA. Equation (18) is sufficiently general for LSA. For CA, using (21), we set $\boldsymbol{N} = \boldsymbol{D}_r^{-\frac{1}{2}}$, $\boldsymbol{L} = \boldsymbol{P} - \boldsymbol{E}$, and $\boldsymbol{G} = \boldsymbol{D}_c^{-\frac{1}{2}}$. This shows that the matrix decomposed in CA in (21) can be formulated in the LSA framework in (18).

The decomposition used in LSA leads to orthonormal matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ used for coordinates; see (20), whereas in CA the decomposition leads to weighted orthonormal matrices $\boldsymbol{\Phi}$ and $\boldsymbol{\Gamma}$; see (24). If we rewrite (20) as $\boldsymbol{U}^T\boldsymbol{I}\boldsymbol{U} = \boldsymbol{I} = \boldsymbol{V}^T\boldsymbol{I}\boldsymbol{V}$, we see this is a difference between using an identity metric $\boldsymbol{I}$ and a metric defined by the margins that are collected in $\boldsymbol{D_r}$ and in $\boldsymbol{D_c}$. The influence of this metric used in CA is most clearly visible in the definition of the chi-squared distances (8), that makes that, for example, for row profiles i and i', equally large differences between columns j and j' are weighted by the margins of j and j' in such a way that a column with a smaller margin takes a larger part in the chi-squared distance between i and i'.

5. Text categorization

LSA is widely used in text categorization (Zhang *et al.*, 2011; Elghazel *et al.*, 2016; Dzisevič and Šešok, 2019; Phillips *et al.*, 2021). However, to our best knowledge, few papers on text categorization use CA, even though CA is similar to LSA. In this section, we compare the performance of LSA and CA in text categorization of three English datasets: BBCNews, BBCSport, and 20 Newsgroups. These datasets have recently been studied in the evaluation of text categorization, for example, Barman and Chowdhury (2020).

5.1 Datasets and methods

The BBCNews dataset (Greene and Cunningham, 2006) consists of 2225 documents that are divided into five categories: "Business" (510 documents), "Entertainment" (386), "Politics" (417), "Sport" (511), and "Technology" (401). The BBCSport dataset (Greene and Cunningham, 2006) consists of 737 documents that are divided into five categories: "athletics" (101), "cricket" (124), "football" (265), "rugby" (147), and "tennis" (100). The 20 Newsgroups dataset, that is, the 20news-bydata version (Rennie, 2005), consists of 18,846 documents that are divided into 20 categories. The dataset is sorted into a training (60%) and a test set (40%). We use a subset of these documents. Specifically, we choose 2963 documents from three categories: "comp.graphics" (584 documents for training set and 389 documents for test set), "rec.sport.hockey" (600 and 399), and "sci.crypt" (595 and 396). The reason we choose a subset (three categories) of 20 Newsgroups is

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

that we want to explore text categorization for datasets with a different but similar number of categories: six (for *Wilhelmus* dataset in Section 6), five (for BBCNews), four (for BBCSport), and three (for a subset of 20 Newsgroups).

To preprocess these datasets, we project all characters to lower case, remove punctuation marks, numbers, and stop words, and apply lemmatization. Subsequently, terms with frequencies lower than 10 are ignored. In addition, following Silge and Robinson (2017), we remove unwanted parts of the 20 Newsgroups dataset such as headers (including fields like "From:" or "Reply-To:" that describe the message), because these are mostly irrelevant for text categorization.

We use two approaches to compare LSA and CA. One is visualization, where we use LSA and CA to visualize documents by projecting them onto two dimensions. The other is to use distance measures to quantitatively evaluate and compare performance in text categorization. We use four different methods based on Euclidean distance for measuring the distance from a document to a set of documents (Guthrie, 2008; Koppel and Seidman, 2013; Kestemont *et al.*, 2016). We choose the Euclidean distance because it plays a central role in the geometric interpretation of LSA and CA (see Sections 2 and 3).

Centroid Euclidean distance between the document and the centroid of the set of documents. The centroid for a set of documents is calculated by averaging the coordinates across all these documents.

In the other three methods, we first calculate the Euclidean distance between the document and every document of the set of documents.

Average average of these Euclidean distances.

Single the minimum Euclidean distance among the Euclidean distances.

Complete the maximum Euclidean distance among the Euclidean distances.

These four methods are similar to the procedures of measuring the distance between clusters in hierarchical clustering analysis, using the centroid, average, single, and complete linkage method, respectively (Jarman, 2020).

In line with the foregoing sections, we denote the raw document-term matrix by $\boldsymbol{F}$. In the case of LSA, we examine four versions: LSA of $\boldsymbol{F}$ (LSA-RAW), LSA of the row-normalized matrices $\boldsymbol{F}^{L1}$ (LSA-NROWL1) and $\boldsymbol{F}^{L2}$ (LSA-NROWL2), and LSA of the TF-IDF matrix $\boldsymbol{F}^{\text{TF-IDF}}$ (LSA-TFIDF). In addition, we also compare performance with the raw document-term matrix, denoted as RAW, where no dimensionality reduction has taken place.

5.2 Visualization

The 2225 documents of the BBCNews dataset lead to a document-term matrix of size 2225 $\times$ 5050. Figure 3 shows the results of an analysis of this document-term matrix by the four LSA methods (LSA-RAW, LSA-NROWL1, LSA-NROWL2, and LSA-TFIDF) and CA. On this dataset, we find that, although the percentage of the total sum of squared singular values in the first two dimensions for CA is lower than the four LSA methods, the four LSA methods do not separate the classes well but CA does a reasonably good job. This is because the margins play an important role in the first two dimensions for the four LSA methods and the relations between documents are blurred by these margins.

The 737 documents of BBCSport dataset lead to a document-term matrix of size 737 $\times$ 2071. Figure 4 shows the results of an analysis of this document-term matrix. Again, we find that the LSA methods do not separate the classes well, but CA does a reasonably good job.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

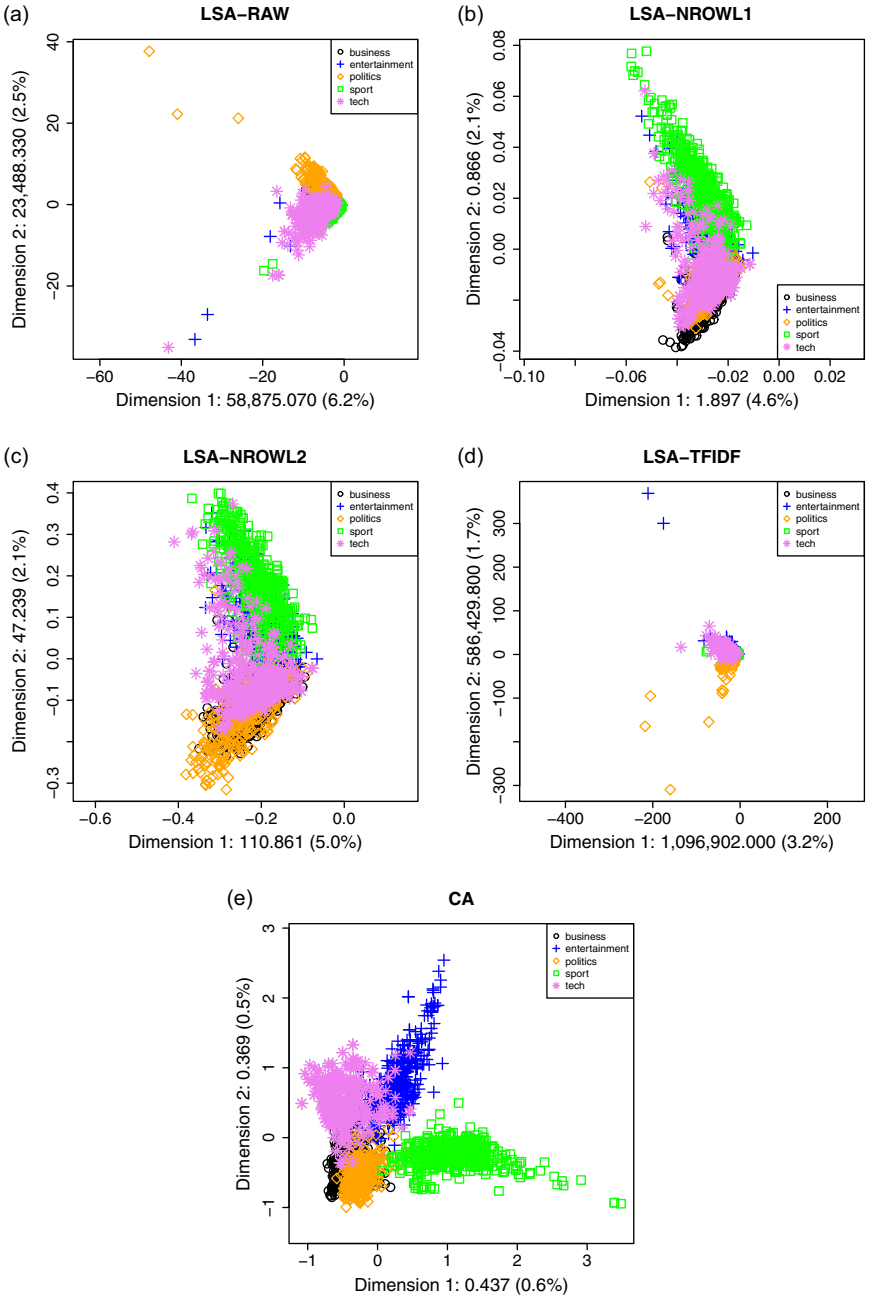

Figure 3. The first two dimensions for each document of BBCNews dataset by (a) LSA-RAW; (b) LSA-NROWL1; (c) LSA-NROWL2; (d) LSA-TFIDF; and (e) CA.

The 2963 documents of 20 Newsgroups dataset lead to a document-term matrix of size 2927 × 2834.[a] Figure 5 shows the results of an analysis of this document-term matrix. On this dataset, we find that CA is doing a reasonably good job, and so do LSA-NROWL1 and LSA-NROWL2.

[a]After preprocessing, 36 documents out of 2963 became empty documents and were removed.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

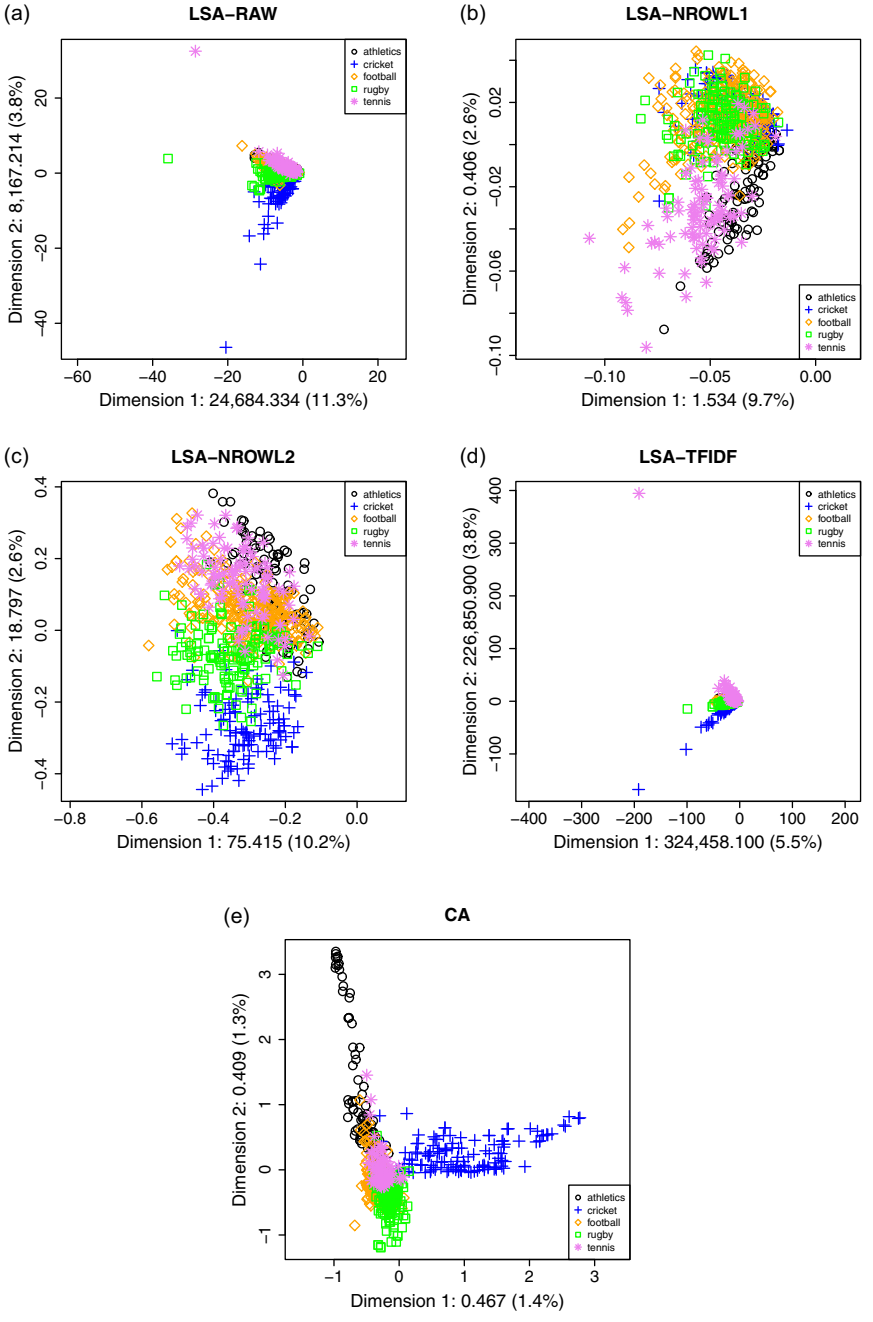

Figure 4. The first two dimensions for each document of BBCSport dataset by (a) LSA-RAW; (b) LSA-NROWL1; (c) LSA-NROWL2; (d) LSA-TFIDF; and (e) CA.

5.3 Distance measures

For the 20 Newsgroups dataset, there is a training and a test set, and we assess the accuracy as a measure for the correct classification of the documents of the test set. For the 20 Newsgroups dataset, there are four steps. First, we apply all four varieties of LSA and CA to all documents

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

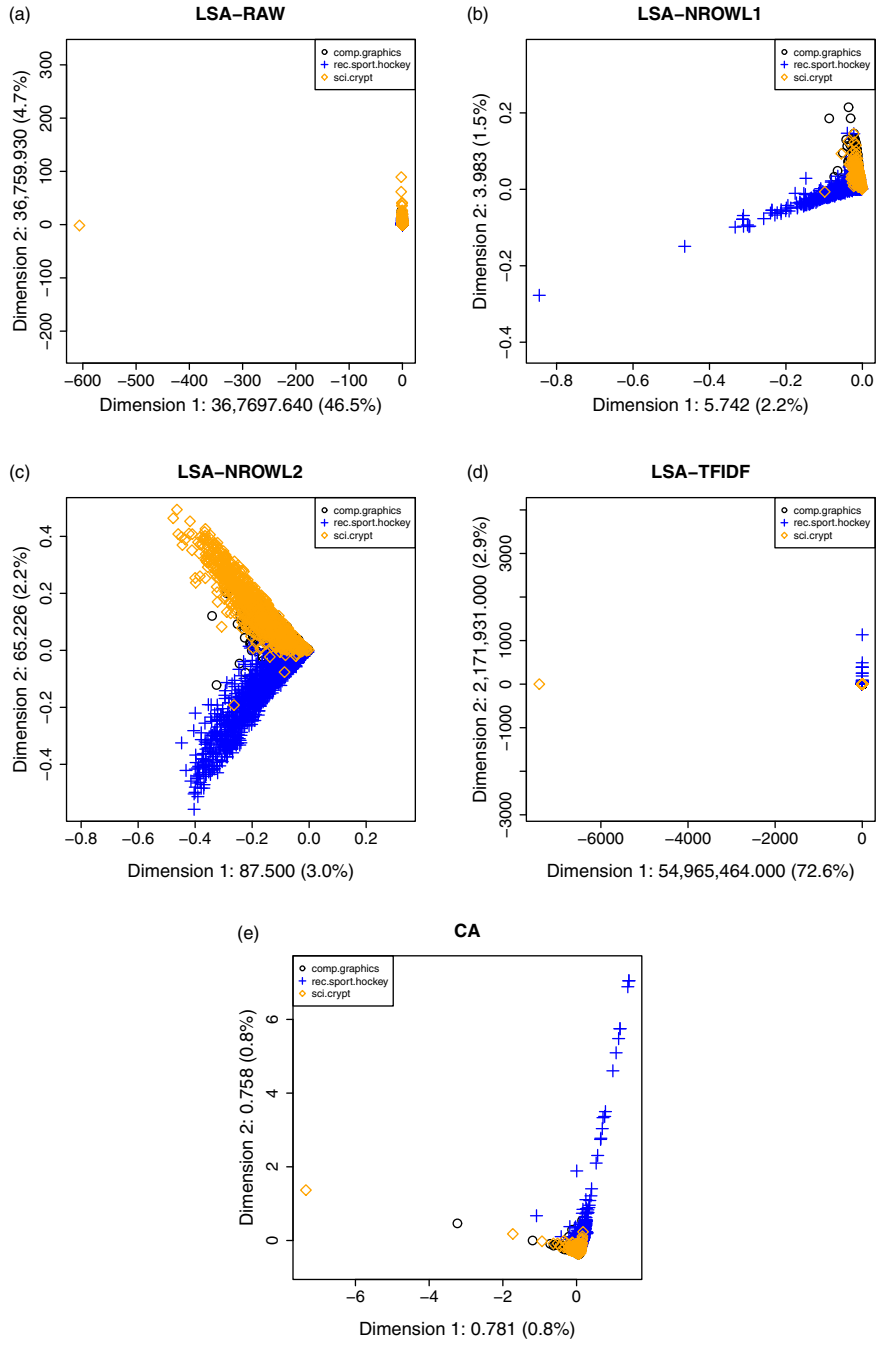

Figure 5. The first two dimensions for each document of 20 Newsgroups dataset by (a) LSA-RAW; (b) LSA-NROWL1; (c) LSA-NROWL2; (d) LSA-TFIDF; and (e) CA.

of the training set. The documents of the test set are projected into the reduced dimensional space; see Sections 2.2.4 and 3. Second, using the centroid, average, single, and complete method, for each document of the test set, the distance between the document and a set of documents for each of three categories ("comp.graphics," "rec.sport.hockey," and "sci.crypt") in the training

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Table 8. The minimum optimal dimensionality k and the accuracy in k for LSA-RAW, LSA-NROWL1, LSA-NROWL2, LSA-TFIDF, and CA, and the accuracy (Acc) for RAW using different distance measurement methods with the BBCNews, BBCSport, and 20 Newsgroups datasets

		Centroid		Average		Single		Complete	
Datasets	Methods	k	Acc	k	Acc	k	Acc	k	Acc
BBCNews	RAW		0.921		0.339		0.791		0.229
	LSA-RAW	401	0.921	7	0.714	24	0.942	1	0.237
	LSA-NROWL1	339	0.947	5	0.898	30	0.948	5	0.723
	LSA-NROWL2	385	0.950	23	0.930	450	0.951	5	**0.829**
	LSA-TFIDF	381	0.942	13	0.725	32	0.953	13	0.253
	CA	318	**0.970**	5	**0.943**	22	**0.961**	4	0.647
BBCSport	RAW		0.917		0.418		0.852		0.193
	LSA-RAW	72	0.919	9	0.843	33	0.930	9	0.332
	LSA-NROWL1	275	0.950	10	0.928	129	0.946	5	0.613
	LSA-NROWL2	96	0.952	103	**0.950**	175	0.955	5	**0.873**
	LSA-TFIDF	486	0.931	9	0.806	20	0.970	7	0.241
	CA	565	**0.978**	24	0.936	35	**0.982**	4	0.420
20Newsgroups	RAW		0.647		0.330		0.688		0.328
	LSA-RAW	214	0.648	9	0.409	26	0.847	2	0.342
	LSA-NROWL1	358	0.897	4	0.847	306	0.852	83	0.412
	LSA-NROWL2	357	0.857	54	0.885	6	0.858	3	**0.735**
	LSA-TFIDF	201	0.617	1	0.347	70	0.863	1	0.340
	CA	84	**0.908**	7	**0.888**	27	**0.902**	11	0.465

Bold values indicate for each dataset the highest accuracy for each distance measurement; underlined values for each dataset the highest accuracy overall.

set is computed. The predicted category for the document is the category with the smallest distance. Third, we compare the predicted category with the true category of the document. Finally, the accuracy is the proportion of correct classifications of all documents of the test set. For BBCNews and BBCSport datasets, in order to evaluate LSA methods and CA, we use five-fold cross-validation (Gareth *et al.*, 2021). That is, the dataset is randomly divided into five folds. The four folds (80% of the dataset) are used as training set, and the remaining one fold (20% of the dataset) is as validation set. The accuracy of each fold is obtained as in the 20 Newsgroups dataset. Then the accuracy is averaged across five folds.

For each form of LSA and for CA, there is an accuracy for each number of dimensions (for five-fold cross-validation, the accuracy is averaged across five folds). The maximum accuracy is the maximum value across these accuracies. Table 8 shows the maximum accuracy for LSA-RAW, LSA-NROWL1, LSA-NROWL2, LSA-TFIDF, and CA for the four distance measures,[b] along with the *minimum* optimal dimension k where this maximum accuracy is reached.[c] First, if we ignore

[b] For BBCSport dataset, we explore the number of all dimensions of dimensionality reduction methods. For BBCNews and 20 Newsgroups datasets, we vary the number of dimension k from 1 to 450.

[c] There is not one single optimal number of dimensions that provides the maximum accuracy; for reasons of space, we show only the lowest in Tables 8 and 9.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

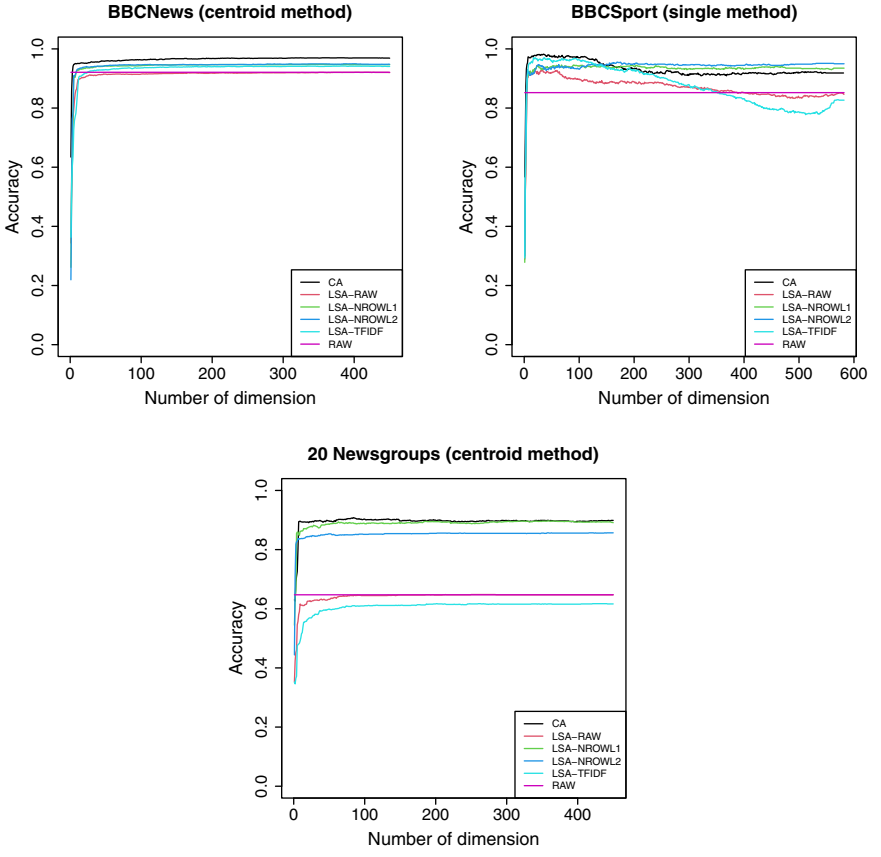

Figure 6. Accuracy as a function of dimension for CA, LSA-RAW, LSA-NROWL1, LSA-NROWL2, LSA-TFIDF, and RAW.

the complete distance method, considering that it has low accuracy overall, CA yields the maximum accuracy compared to the RAW method (i.e., without dimensionality reduction) as well as all four LSA methods for each combination of dataset and other distance measurement method, except for the BBCSport dataset with the average method, where CA has the second largest accuracy. Second, for each dataset, CA is doing best overall. Specifically, CA with the centroid, the single, and the centroid distance method provides the best accuracy for BBCNews, BBCSport, and 20 Newsgroups datasets, respectively.

In order to further explore different dimensionality reduction methods under optimal distance measurement method which provides highest accuracy, Figure 6 shows the accuracy as a function of the numbers of dimensions under centroid, single, and centroid methods for BBCNews, BBCSport, and 20 Newsgroups datasets, respectively. CA in combination with the optimal distance measurement method performs better than the other methods over a large range, especially for BBCNews dataset, almost irrespective of dimension.

6. Authorship attribution

In this section, we examine the performance of LSA and CA on a dataset originally set up for authorship attribution. We first use the dataset to see how well LSA and CA are able to assign documents with a known author to the correct author. Second, we assign a document with unknown author to one of the known authors.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Authorship attribution is the process of identifying the authorship of a document; its applications include plagiarism detection and resolving of authorship disputes (Bozkurt, Baghoglu, and Uyar, 2007) and are particularly relevant for historical texts, where other historical records are not sufficient to determine authorship. Both LSA and CA have been used for authorship attribution before. For example, Soboroff *et al.* (1997) applied LSA with n-grams as terms to visualize authorship among biblical Hebrew texts. McCarthy *et al.* (2006) applied LSA to lexical features to automatically detect semantic similarities between words (Stamatatos, 2009). Satyam *et al.* (2014) used LSA on a character n-gram-based representation to build a similarity measure between a questioned document and known documents. Mealand (1995) studied the Gospel of Luke using a visualization provided by CA. Mealand (1997) also measured genre differences in Mark by CA. Mannion and Dixon (2004) applied CA to study authorship attribution of the case of Oliver Goldsmith by visualization.

The *Wilhelmus* is the national anthem of the Netherlands, and its authorship is unknown and much debated. There is a substantive amount of qualitative research attempting to determine the authorship of the *Wilhelmus*, with quantitative or statistical methods being used relatively recently. To the best of our knowledge, the authorship of the *Wilhelmus* was first studied by statistical methods and computational means in Winkel (2015), whose results on authorship attribution were inconclusive. After that, Kestemont *et al.* (2017a, 2017b) studied the question using principal component analysis and the General Imposters (GI) method, attributing the *Wilhelmus* to the writer Datheen. Vargas Quiros (2017) used the data of Kestemont *et al.* (2017a, 2017b) and applied the KRIMP compression algorithm (Van Leeuwen, Vreeken, and Siebes, 2006) and Kullback–Leibler Divergence—they tended to agree with Kestemont *et al.* (2017a, 2017b), even though the KRIMP attributed the *Wilhelmus* to another author when a different feature selection method was used. Thus, the results were inconclusive, with a tendency to prefer Datheen. Our paper provides further evidence in favor of attributing the authorship to *Datheen*.

6.1 Data and methods

We use a total of 186 documents by 6 writers, consisting of 35 documents written by Datheen, 46 by Marnix, 23 by Heere, 35 by Haecht, 33 by Fruytiers, and 14 by Coornhert. These documents contain tag-lemma pairs as terms, obtained through part-of-speech tagging and lemmatizing of the texts, and are made publicly available by Kestemont *et al.* (2016, 2017a, 2017b). The average marginal frequencies range from 406 for documents by Fruytiers to 545 for documents by Haecht. See Kestemont (2017) for more details regarding the dataset. Similar to Section 5, in this section, we also use visualization and distance measures to compare LSA and CA.

6.2 Visualization

We first examine all documents of two authors Marnix and Datheen,[d] using the 300 most frequent tag-lemma pairs. These form a document-term matrix of size 81 × 300. Figure 7 shows the results of analyzing this document-term matrix using the four LSA methods (LSA-RAW, LSA-NROWL1, LSA-NROWL2, and LSA-TFIDF), and CA. The *Wilhelmus* document is not included in the data matrix, but it is projected into the solutions for illustrative purposes by W, in red; see Sections 2.2.4 and 3. As seen in Figure 7, all four varieties of LSA fail to show a clear separation, while CA separates documents by the two authors clearly, even though the first two dimensions for CA account for a much smaller percentage of the total sum of squared singular values than the first two dimensions for the four LSA methods. This is because the margins play an important role

[d]We chose these two authors specifically, out of our dataset, as they are the two main contenders for the authorship of *Wilhelmus*—Marnix has been the most popular candidate from qualitative analysis, and since the work of Kestemont *et al.*, (2017a, 2017b) Datheen is also a serious candidate.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

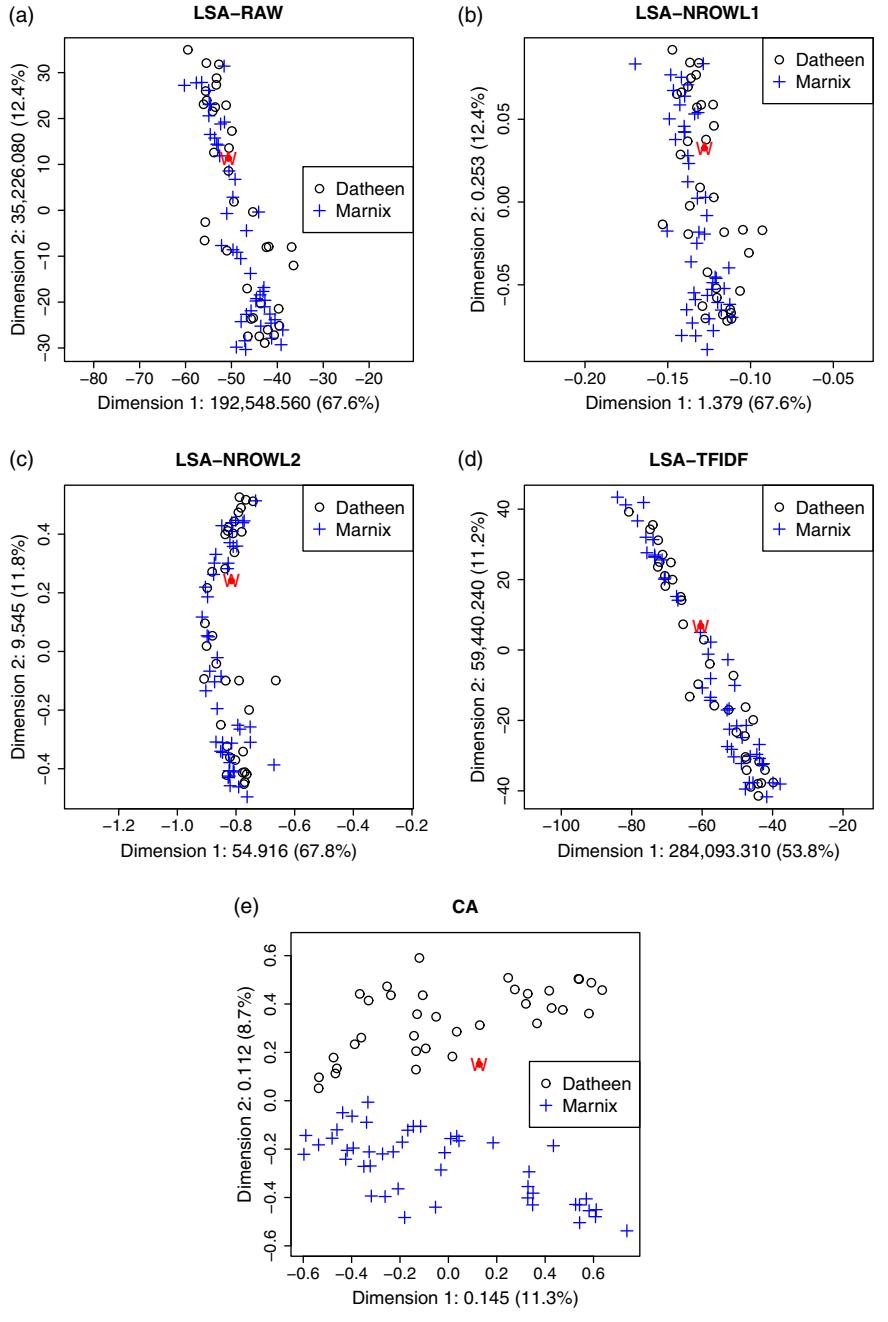

Figure 7. The first two dimensions for each document of author Datheen and author Marnix, and the *Wilhelmus* (in red) by (a) LSA-RAW; (b) LSA-NROWL1; (c) LSA-NROWL2; (d) LSA-TFIDF; and (e) CA.

in the first two dimensions for the four LSA methods and the relations between documents are blurred by these margins. We also see that in CA the *Wilhelmus* is clearly attributed to Datheen.

Given the effectiveness of CA and the attribution of the *Wilhelmus* to Datheen in the above analysis, we now show visualizations of CA for documents by Datheen and four other authors in turn (Figure 8). For three out of four authors, there is a clear separation between that author and

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

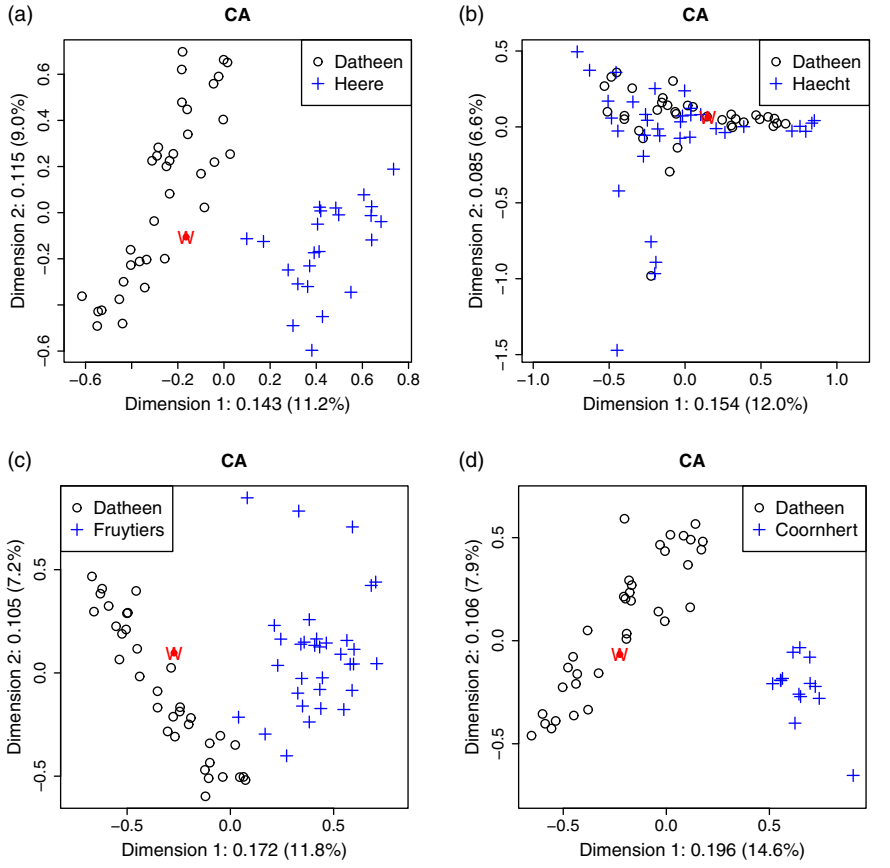

Figure 8. The first two dimensions for each document of author Datheen and another author, and the *Wilhelmus* (in red) using CA: (a) Heere; (b) Haecht; (c) Fruytiers; and (d) Coornhert.

Datheen. In the case Haecht however (sub-figure (b)), there is no clear separation from Datheen. In all three cases where there is a clear separation, *Wilhelmus* is attributed to Datheen, as before.

Finally, we apply all four varieties of LSA and CA to all documents of the six authors, which form a document-term matrix of size 186×300. Figure 9 shows the results of the analysis of this matrix by LSA-RAW, LSA-NROWL1, LSA-NROWL2, LSA-TFIDF, and CA. The *Wilhelmus* is projected into the solutions afterward. Again we find that, although the percentage of the total sum of squared singular values in the first two dimensions for CA is lower than the four LSA methods, CA separates the documents quite well compared with the four LSA methods. For instance, documents written by Marnix are effectively separated from the documents written by other authors. The documents of the other authors also seem to form much more distinguishable clusters, as compared to LSA, except for Datheen and Haecht.

6.3 Distance measures

To evaluate LSA methods and CA, we use leave-one-out cross-validation (LOOCV) (Gareth *et al.*, 2021) with the 186 documents of 6 authors. Using LOOCV, each time we discern the following four steps. At the first step, a single document of the 186 documents is used as the validation set and the remaining 185 documents make up the training set. The 185 documents of training set form a document-term matrix with 185 rows and 300 columns. At step two, we perform LSA-RAW,

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

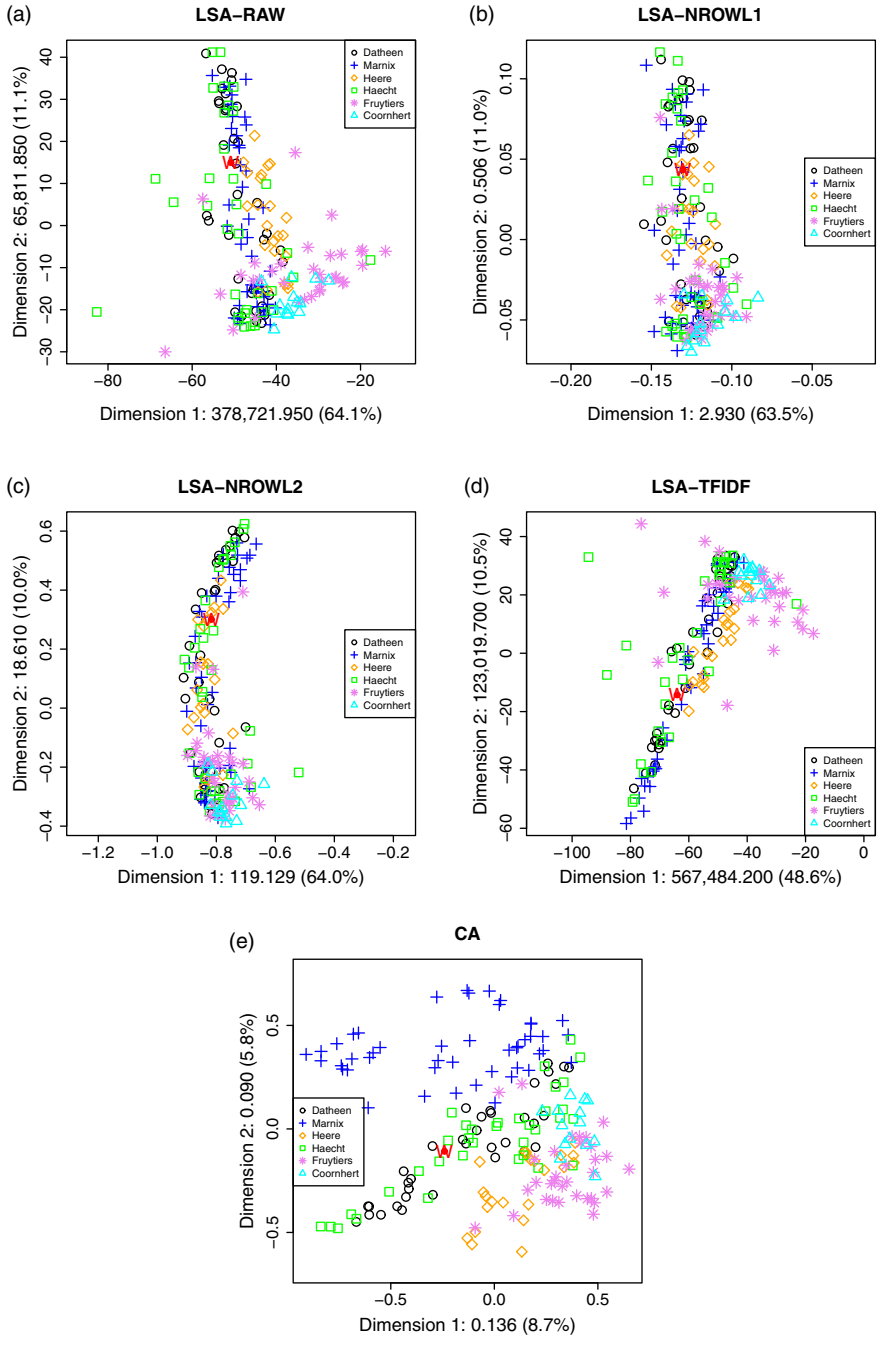

Figure 9. The first two dimensions for each document of six authors, and the *Wilhelmus* (in red) by (a) LSA-RAW; (b) LSA-NROWL1; (c) LSA-NROWL2; (d) LSA-TFIDF; and (e) CA.

LSA-NROWL1, LSA-NROWL2, LSA-TFIDF, and CA on this document-term matrix to obtain the coordinates of the 185 documents. The single document of validation set is projected into the solutions, see Sections 2.2.4 and 3. At step three, using the centroid, average, single, and complete method, the distance is computed between the single document and the six author groups of

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Table 9. The minimum optimal dimensionality *k* and the accuracy in *k* for LSA-RAW, LSA-NROWL1, LSA-NROWL2, LSA-TFIDF, and CA, and the accuracy for RAW using different distance measurement methods with *Wilhelmus* dataset

Methods	Centroid		Average		Single		Complete	
Methods	*k*	Accuracy	*k*	Accuracy	*k*	Accuracy	*k*	Accuracy
RAW		0.720		0.522		0.672		0.177
LSA-RAW	51	0.720	70	0.554	14	0.720	1	0.296
LSA-NROWL1	93	0.731	116	0.645	22	0.710	75	0.226
LSA-NROWL2	59	0.742	41	0.699	21	0.715	77	0.301
LSA-TFIDF	84	0.720	90	0.538	23	0.731	1	0.231
CA	151	**0.930**	12	**0.790**	19	**0.785**	95	**0.452**

Bold values indicate the highest accuracy for each distance measurement; the underlined value indicates the highest accuracy overall.

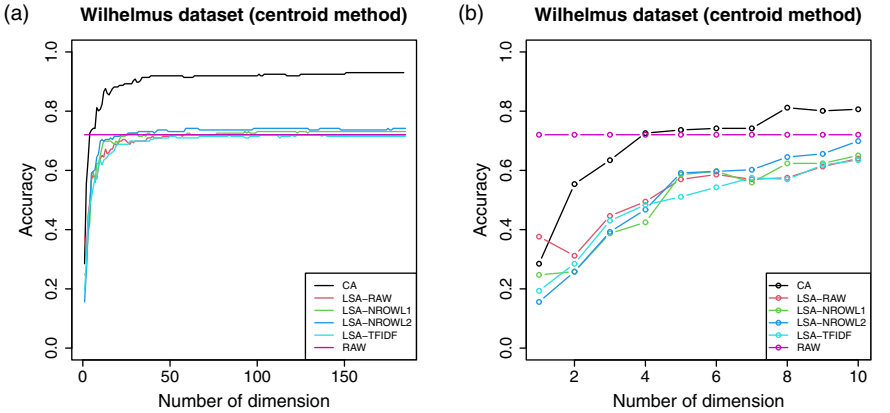

Figure 10. Accuracy versus the number of dimensions (centroid method) for CA, RAW, LSA-RAW, LSA-NROWL1, LSA-NROWL2, and LSA-TFIDF with *Wilhelmus* dataset.

documents. For this single document, the predicted author of the document is the author with the smallest distance. At the final step, we compare the predicted author with the true author of the single document. We repeat this 186 times, once for each single document. The accuracy is calculated by the ratio: number of times an author is correctly predicted divided by 186.

Table 9 shows the maximum accuracy for LSA-RAW, LSA-NROWL1, LSA-NROWL2, LSA-TFIDF, and CA for the four distance measures,[e] along with the minimum optimal dimension *k*. First, CA yields the maximum accuracy for all distance measurement methods as compared to the RAW method as well as all four LSA methods. Second, CA with the centroid method provides the highest accuracy.

In order to further explore the centroid method, Figure 10 shows the accuracy with different numbers of dimensions for LSA-RAW, LSA-NROWL1, LSA-NROWL2, LSA-TFIDF, and CA. Figure 10a displays all dimensions on the horizontal axis, and Figure 10b focuses on the first 10 dimensions. CA in combination with the centroid method performs better than the other methods almost irrespective of dimension, except for the very first ones. Also, the accuracy of CA in combination with the centroid method is very high over a large range.

[e]For *Wilhelmus* dataset, we explore the number of all dimensions of dimensionality reduction methods.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

6.4 Authorship attribution of the Wilhelmus

Since CA in combination with the centroid method appears to be the best overall, we use them to determine the authorship of the *Wilhelmus*. In the 34 optimal dimensions (dimensions 151–184), we find that the *Wilhelmus* is attributed to the author Datheen, while Haecht is the second most likely candidate. The distance of the *Wilhelmus* to the centroid of documents of Datheen averaged across 34 optimal dimensions is 0.825, to Haecht 0.880, to Marnix 0.939, to Heere 1.015, to Fruytiers 1.064, and to Coornhert 1.253. Thus, CA attributes *Wilhelmus* to Datheen and provides more weight using an independent statistical technique, to prior results by Kestemont *et al.* (2017a, 2017b) in resolving this debate.

7. Conclusion

LSA and CA both allow for dimensionality reduction by the SVD of a matrix; however, the actual matrix analyzed by LSA and CA is different, and therefore LSA and CA capture different kinds of information. In LSA, we apply an SVD to $\boldsymbol{F}$, or to a weighted $\boldsymbol{F}$. In CA, an SVD is applied to the matrix $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ of standardized residuals. The elements in $\boldsymbol{D}_r^{-\frac{1}{2}}(\boldsymbol{P}-\boldsymbol{E})\boldsymbol{D}_c^{-\frac{1}{2}}$ display the departure from the margins, that is, departure from the expected frequencies under independence collected in $\boldsymbol{E}$. Due to $\boldsymbol{E}$, in CA the effect of the margins is eliminated—a solution only displays the dependence between documents and terms. Concluding, in LSA, the effect of the margins as well as the dependence is part of the matrix that is analyzed, and these margins usually play a dominant role in the first dimension of the LSA solution as usually on the first dimension all points depart in the same direction from the origin. On the other hand, in CA all points are scattered around the origin and the origin represents the profile of the row and column margins of $\boldsymbol{F}$.

In summary, although LSA allows a study of the relations between documents, between terms, and between documents and terms, this study is not easy. The reason is that these relations are blurred by the effect of the margins that are also displayed in the LSA solution. CA does not have this property. Therefore, it appears that CA is a better tool for studying the relations between documents, between terms, and between documents and terms. Also, discussed in Section 3, CA has many nice properties like providing a geometric display where the Euclidean distances approximate the χ^2-distances between the rows and between the columns of the matrix, and the relation to the Pearson χ^2 statistic. Overall, from a theoretical point of view, it appears that CA has more attractive properties than LSA. Empirically, we evaluated and compared the two methods on text categorization in English and authorship attribution in Dutch and found that CA can both separate documents better visually and obtain higher accuracies on text categorization and authorship attribution as compared to LSA techniques.

A document-term matrix is similar to a word-context matrix, commonly used to represent word meanings, in the sense that it is also a matrix of counts. However, in the context of word-context matrices the ways in which the counts are transformed are usually different from the way they are transformed for document-term matrices, and therefore, due to space limitations, we defer a comparison of CA and LSA of word-context matrices to future work. In the future, it is also interesting to compare word embeddings learned by LSA-based methods and CA to more recent static word embedding approaches such as Word2Vec and GloVe, or even contextualized word embeddings models like BERT. And it is interesting to compare LSA-based methods and CA on recent classifiers, such as neural network models.

Acknowledgments. Author Qianqian Qi is supported by the China Scholarship Council.

Competing interests declaration. Author Qianqian Qi is supported by the China Scholarship Council. Author David J. Hessen, Author Tejaswini Deoskar, and Author Peter G. M. van der Heijden declare none.

References

Ab Samat N., Murad M. A. A., Abdullah M. T. and Atan R. (2008). Term weighting schemes experiment based on SVD for Malay text retrieval. *International Journal of Computer Science and Network Security (IJCSNS)* **8**(10), 357–361.

Aggarwal C. C. (2018). *Machine Learning for Text*. Cham: Springer.

Albright R. (2004). *Taming Text with the SVD*. North Carolina, USA: SAS Institute Inc.

Barman D. and Chowdhury N. (2020). A novel semi supervised approach for text classification. *International Journal of Information Technology* **12**(4), 1147–1157.

Benzécri J.-P. (1973). *L'analyse des données*, 1 and 2. Paris, France: Dunod.

Berry M. W., Dumais S. T. and O'Brien G. W. (1995). Using linear algebra for intelligent information retrieval. *SIAM Review* **37**(4), 573–595.

Bozkurt I. N., Baghoglu O. and Uyar E. (2007). *Authorship attribution*. 2007 22nd International Symposium on Computer and Information Sciences, Ankara, Turkey, pp. 1–5.

Deerwester S., Dumais S. T., Furnas G. W., Landauer T. K. and Harshman R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science* **41**(6), 391–407.

Deisenroth M. P., Faisal A. A. and Ong C. S. (2020). *Mathematics for Machine Learning*. Cambridge: Cambridge University Press.

Di Gangi M. A., Bosco G. L. and Pilato G. (2019). Effectiveness of data-driven induction of semantic spaces and traditional classifiers for sarcasm detection. *Natural Language Engineering* **25**(2), 257–285.

Dumais S. T. (1991). Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments, & Computers* **23**(2), 229–236.

Dumais S. T., Furnas G. W., Landauer T. K., Deerwester S. and Harshman R. (1988). *Using latent semantic analysis to improve access to textual information*. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Washington, DC, USA, pp. 281–285.

Dzisevič R. and Šešok D. (2019). *Text classification using different feature extraction approaches*. 2019 Open Conference of Electrical, Electronic and Information Sciences (eStream), Vilnius, Lithuania, pp. 1–4.

Elghazel H., Aussem A., Gharroudi O. and Saadaoui W. (2016). Ensemble multi-label text categorization based on rotation forest and latent semantic indexing. *Expert Systems with Applications* **57**, 1–11.

Frobenius G. (1912). Über matrizen aus nicht negativen elementen.

Gareth J., Daniela W., Trevor H. and Robert T. (2021). *An Introduction to Statistical Learning: With Applications in R*. New York: Springer.

Gifi A. (1990). *Nonlinear Multivariate Analysis*. Chichester: Wiley.

Greenacre M. J. (1984). *Theory and Applications of Correspondence Analysis*. London: Academic Press.

Greenacre M. J. (2017). *Correspondence Analysis in Practice*. Boca Raton, FL: CRC Press.

Greenacre M. J. and Hastie T. (1987). The geometric interpretation of correspondence analysis. *Journal of the American Statistical Association* **82**(398), 437–447.

Greene D. and Cunningham P. (2006). *Practical solutions to the problem of diagonal dominance in kernel document clustering*. Proceedings of the 23rd International Conference on Machine Learning, New York, NY, USA, pp. 377–384.

Gupta H. and Patel M. (2021). *Method Of Text Summarization Using Lsa And Sentence Based Topic Modelling With Bert*. 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, pp. 511–517.

Guthrie D. (2008). Unsupervised Detection of Anomalous Text, PhD Thesis, Department of Computer Science, University of Sheffield

Hassani A., Iranmanesh A. and Mansouri N. (2021). Text mining using nonnegative matrix factorization and latent semantic analysis. *Neural Computing and Applications* **33**(20), 13745–13766.

Hayashi C. (1956). Theory and example of quantification (II). *Proceedings of the Institute of Statistical Mathematics* **4**, 19–30.

Hayashi C. (1992). Quantification method III or correspondence analysis in medical science. *Annals of Cancer Research and Therapy* **1**(1), 17–21.

Hill M. O. (1973). Reciprocal averaging: An eigenvector method of ordination. *Journal of Ecology* **61**(1), 237–249.

Hill M. O. (1974). Correspondence analysis: A neglected multivariate method. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* **23**(3), 340–354.

Hou R. and Huang C.-R. (2020). Classification of regional and genre varieties of Chinese: A correspondence analysis approach based on comparable balanced corpora. *Natural Language Engineering* **26**(6), 613–640.

Hu X., Cai Z., Franceschetti D., Penumatsa P., Graesser A., Louwerse M., McNamara D. S. and Tutoring Research Group (2003). LSA: First dimension and dimensional weighting, *Proceedings of the Annual Meeting of the Cognitive Science Society*, Boston, USA, **25**.

Jarman A. M. (2020). *Hierarchical Cluster Analysis: Comparison of Single Linkage, Complete Linkage, Average Linkage and Centroid Linkage Method*. Georgia, USA: Georgia Southern University.

Jiao Q. and Zhang S. (2021). *A brief survey of word embedding and its recent development*. 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, **5**, pp. 1697–1701.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Jurafsky D. and Martin J. H. (2021). Speech and language processing, (3rd ed. draft), chapter 6. Retrieved October 20, 2022, from https://web.stanford.edu/jurafsky/slp3/.

Kalmukov Y. (2022). Comparison of latent semantic analysis and vector space model for automatic identification of competent reviewers to evaluate papers. *International Journal of Advanced Computer Science and Applications* **13**(2), 77–85.

Kestemont M. (2017). Who wrote the Wilhelmus? Retrieved July 17, 2021, from https://github.com/mikekestemont/anthem.

Kestemont M., **Stover J.**, **Koppel M.**, **Karsdorp F. and Daelemans W.** (2016). Authenticating the writings of Julius Caesar. *Expert Systems with Applications* **63**, 86–96.

Kestemont M., **Stronks E.**, **De Bruin M. and Winkel T.D** (2017a). *Did a poet with donkey ears write the oldest anthem in the world? Ideological implications of the computational attribution of the Dutch national anthem to Petrus Dathenus*. Digital Humanities 2017, Conference Abstracts, Montreal, Canada.

Kestemont M., **Stronks E.**, **De Bruin M. and Winkel T.D** (2017b). *Van wie is het Wilhelmus? De auteur van het Nederlandse volkslied met de computer onderzocht*. Amsterdam: Amsterdam University Press.

Kolda T. G. and O'leary D. P. (1998). A semidiscrete matrix decomposition for latent semantic indexing information retrieval. *ACM Transactions on Information Systems* **16**(4), 322–346.

Koppel M. and Seidman S. (2013). *Automatically identifying pseudepigraphic texts*. Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, Seattle, Washington, USA, pp. 1449–1454.

Landauer T. K. and Dumais S. T. (1997). A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review* **104**(2), 211–240.

Levy O. and Goldberg Y. (2014). Neural word embedding as implicit matrix factorization, *Advances in Neural Information Processing Systems*, vol. 27. New York, USA: Curran Associates, Inc.

Levy O., **Goldberg Y. and Dagan I.** (2015). Improving distributional similarity with lessons learned from word embeddings. *Transactions of the Association for Computational Linguistics* **3**, 211–225.

Mannion D. and Dixon P. (2004). Sentence-length and authorship attribution: The case of Oliver Goldsmith. *Literary and Linguistic Computing* **19**(4), 497–508.

McCarthy P. M., **Lewis G. A.**, **Dufty D. F. and McNamara D. S.** (2006). *Analyzing writing styles with Coh-Metrix*. Proceedings of the Nineteenth International Florida Artificial Intelligence Research Society Conference, Melbourne Beach, FL, USA, pp. 764–769.

Mealand D. L. (1995). Correspondence analysis of Luke. *Literary and Linguistic Computing* **10**(3), 171–182.

Mealand D. L. (1997). Measuring genre differences in Mark with correspondence analysis. *Literary and Linguistic Computing* **12**(4), 227–245.

Michailidis G. and De Leeuw J. (1998). The Gifi system of descriptive multivariate analysis. *Statistical Science* **13**(4), 307–336.

Morin A. (1999). Knowledge extraction in texts: A comparison of two methods. Retrieved July 17, 2021, from https://www.stat.fi/isi99/proceedings/arkisto/varasto/mori0673.pdf.

Nakov P., **Popova A. and Mateev P.** (2001). *Weight functions impact on LSA performance*. EuroConference Recent Advances in Natural Language Processing, Bulgaria: Tzigov Chark, pp. 187–193.

Patil A. (2022). *Word Significance Analysis in Documents for Information Retrieval by LSA and TF-IDF using Kubeflow*. Expert Clouds and Applications: Proceedings of ICOECA 2021, Singapore: Springer, pp. 335–348.

Perron O. (1907). Zur theorie der matrices. *Mathematische Annalen* **64**(2), 248–263.

Phillips T., **Saleh A.**, **Glazewski K. D.**, **Hmelosilver C. E.**, **Lee S.**, **Mott B. and Lester J. C.** (2021). *Comparing natural language processing methods for text classification of small educational data*. Companion Proceedings 11th International Conference on Learning Analytics & Knowledge, Irvine, CA, USA.

Ren X. and Coutanche M. N. (2021). Sleep reduces the semantic coherence of memory recall: An application of latent semantic analysis to investigate memory reconstruction. *Psychonomic Bulletin & Review* **28**(4), 1336–1343.

Rennie J. (2005). 20 newsgroups data set. Retrieved April 21, 2022, from http://qwone.com/jason/20Newsgroups/.

Salton G. and Buckley C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management* **24**(5), 513–523.

Satyam A., **Dawn A. K. and Saha S. K.** (2014). *A statistical analysis approach to author identification using latent semantic analysis: Notebook for pan at clef 2014*. 2014 Working Notes for CLEF Conference, Sheffield, UK.

Séguéla J. and Saporta G. (2011). *A comparison between latent semantic analysis and correspondence analysis*. CARME 2011 International Conference on Correspondence Analysis and Related Methods, Rennes, France.

Séguéla J. and Saporta G. (2013). A hybrid recommender system to predict online job offer performance. *Revue des Nouvelles Technologies de l'Information* **25**, 177–197.

Silge J. and Robinson D. (2017). *Text Mining with R: A Tidy Approach*. Sebastopol, CA: O'Reilly Media.

Soboroff I. M., **Nicholas C. K.**, **Kukla J. M. and Ebert D. S.** (1997). *Visualizing document authorship using n-grams and latent semantic indexing*. Proceedings of the 1997 Workshop on New Paradigms in Information Visualization and Manipulation, New York: NY, USA, pp. 43–48.

Stamatatos E. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology* **60**(3), 538–556.

https://doi.org/10.1017/S1351324923000244 Published online by Cambridge University Press

Tseng H.-C., **Chen B.**, **Chang T.-H. and Sung Y.-T.** (2019). Integrating LSA-based hierarchical conceptual space and machine learning methods for leveling the readability of domain-specific texts. *Natural Language Engineering* **25**(3), 331–361.

Van Dam A., **Dekker M.**, **Morales-Castilla I.**, **Rodríguez M.Á.**, **Wichmann D. and Baudena M.** (2021). Correspondence analysis, spectral clustering and graph embedding: Applications to ecology and economic complexity. *Scientific Reports* **11**(1): 8926. https://doi.org/10.1038/s41598-021-87971-9

Van der Heijden P. G. M., **De Falguerolles A. and De Leeuw J.** (1989). A combined approach to contingency table analysis using correspondence analysis and loglinear analysis. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **38**(2), 249–292.

Van Leeuwen M., **Vreeken J. and Siebes A.** (2006). *Compression picks item sets that matter*. Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases, Berlin, Germany, pp. 585–592.

Vargas Quiros J. (2017). Information-theoretic anomaly detection and authorship attribution in literature, Master's Thesis. Department of Information and Computing Sciences, Utrecht University

Winkel T.d (2015). Of Deutsches blood, Master's Thesis. Utrecht University

Zhang W., **Yoshida T. and Tang X.** (2011). A comparative study of TF∗IDF, LSI and multi-words for text classification. *Expert Systems with Applications* **38**(3), 2758–2765.

Cite this article: Qi Q, Hessen DJ, Deoskar T and van der Heijden PGM. A comparison of latent semantic analysis and correspondence analysis of document-term matrices. *Natural Language Engineering* https://doi.org/10.1017/S1351324923000244