Machine Learning (2024) 113:2183–2205
https://doi.org/10.1007/s10994-023-06424-8

Exploiting sparsity and statistical dependence in multivariate data fusion: an application to misinformation detection for high-impact events

Lucas P. Damasceno[1] · Egzona Rexhepi[2] · Allison Shafer[2] · Ian Whitehouse[2] · Nathalie Japkowicz[2] · Charles C. Cavalcante[1] · Roberto Corizzo[2] · Zois Boukouvalas[2]

Received: 10 March 2023 / Revised: 10 August 2023 / Accepted: 3 October 2023 /
Published online: 29 November 2023

Abstract
With the evolution of social media, cyberspace has become the de-facto medium for users to communicate during high-impact events such as natural disasters, terrorist attacks, and periods of political unrest. However, during such high-impact events, misinformation can spread rapidly on social media, affecting decision-making and creating social unrest. Identifying the spread of misinformation during high-impact events is a significant data challenge, given the *multi-modal data* associated with social media posts. Advances in multi-modal learning have shown promise for detecting misinformation; however, key limitations still make this a significant challenge. These limitations include the *explicit* and *efficient* modeling of the underlying non-linear associations of multi-modal data geared at misinformation detection. This paper presents a novel avenue of work that demonstrates how to frame the problem of misinformation detection in social media using multi-modal latent variable modeling and presents two novel algorithms capable of modeling the underlying associations of multi-modal data. We demonstrate the effectiveness of the proposed algorithms using simulated data and study their performance in the context of misinformation detection using a popular multi-modal dataset that consists of tweets published during several high-impact events.

Keywords Data fusion · Independent vector analysis · Multi-modal learning · Misinformation detection · Deep learning

Editors: Dino Ienco, Roberto Interdonato, Pascal Poncelet.

✉ Lucas P. Damasceno
lucaspdamasceno@alu.ufc.br

✉ Zois Boukouvalas
boukouva@american.edu

[1] Federal University of Ceará, Fortaleza, CE 60455-760, Brazil

[2] American University, Washington, DC 20016, USA

1 Introduction

With the evolution of social media technologies, there has been a fundamental change in how information is accessed, shared, and propagated. Propagation of information, particularly misinformation, becomes especially important during high-impact events such as pandemics, natural disasters, terrorist attacks, and periods of political transition, unrest, or financial instability.

Recent multi-modal learning advances have shown promise for detecting misinformation (Sharma et al., 2019); however, this problem remains a significant challenge due to several key limitations. One limitation relates to the use of multi-modal data, i.e., information collected about the same phenomenon using different modalities. The use of multi-modal data has not been fully leveraged in intelligent systems, which traditionally utilize a single modality, typically text (Moroney et al., 2021) or images (Cao et al., 2020). Machine learning algorithms must be able to understand content holistically to become more effective in detecting misinformation.

Early fusion methods provide effective solutions for multi-modal learning since joint representations of input features from different modalities are created before attempting to classify the content, enabling enhanced detection of posts with malicious content (Baltrušaitis et al., 2018). However, in these studies, the joint representations are obtained by simply concatenating the individual representations or implicitly modeling the mutual relationships across the modalities (Ramachandram & Taylor, 2017). Multi-modal learning is desirable because of its ability to *explicitly* learn the mutual relationships among the modalities by letting multiple sources of information adaptively interact while generating the joint feature representations. This is what we define as *true fusion*.

With its well-structured formulation, independent vector analysis (IVA) provides an ideal starting point for developing methods for true fusion. Through the estimation of joint features, IVA can effectively capture unique characteristics of multi-modal data that can be used to enhance the performance of a machine learning task. The results presented in Boukouvalas et al. (2021) demonstrate this idea and use true fusion to exploit the underlying complementary information contained in different molecular featurization methods demonstrating significant advantages over currently used methods (Boukouvalas et al., 2021; Balakrishnan et al., 2021). In addition, IVA algorithms are computationally attractive and easily interpretable? (Boukouvalas et al., 2021; Damasceno et al., 2022).

This work presents two novel multi-modal IVA-based algorithms geared at detecting misinformation. Our first algorithm, independent vector analysis by multivariate entropy maximization with kernels, or IVA-M-EMK, effectively models complex, non-linear relationships among different modalities. Our second algorithm, independent vector analysis with sparse inverse covariance estimation, or IVA-SPICE, imposes sparsity constraints through the inverse covariance matrix isolating the most important relationships in the underlying multi-modal features. We demonstrate the effectiveness of IVA-M-EMK and IVA-SPICE under different scenarios using simulated data and study their performance in the context of misinformation detection using a popular multi-modal dataset consisting of tweets made during several high-impact events.

It is worth noting that achieving a perfect classification score is not the goal of this work. Instead, our goal is to present a novel avenue of work that will demonstrate the great potential of latent variable methods for true fusion of multi-modal data in a fast-growing and challenging area. In particular, we present how to generate joint features from multi-modal datasets to improve the learned response of a detection model over the performance

of the individual feature vectors treated separately. Last but not least, our algorithms are transferable and fully generalizable to other areas of interest where analysis of multi-modal data is vital.

The remainder of the paper is organized as follows. Section 2 discusses a multi-modal fusion framework based on IVA and presents the mathematical details of IVA-M-EMK and IVA-SPICE. In Sect. 3, we demonstrate the effectiveness of the proposed algorithms using simulated data and present the parallel implementation of IVA-SPICE. In Sect. 4, we justify the importance of explicitly modeling the non-linear relationships across different modalities using IVA-M-EMK and IVA-SPICE. Then, we numerically demonstrate under which scenarios the proposed algorithms enhance the performance of the detection of misinformation using a popular multi-modal dataset consisting of tweets posted during several high-impact events. Finally, Sect. 5 concludes the paper.

2 Multi-modal data fusion framework based on independent vector analysis

We formulate the problem of joint feature generation for detection of unreliable posts as a joint blind source separation (JBSS) problem. In particular, let $\mathbf{X}^{[k]} \in \mathbb{R}^{d\times V}$ is the kth observation matrix from kth modality, where d denotes the number of initial high level feature vectors in the kth modality and V denotes the total number of samples. The model is given by

$$\mathbf{X}^{[k]} = \mathbf{A}^{[k]}\mathbf{S}^{[k]}, \quad k = 1, \ldots, K, \tag{1}$$

where $\mathbf{A}^{[k]} \in \mathbb{R}^{d\times N}$ is the kth mixing matrix and $\mathbf{S}^{[k]} \in \mathbb{R}^{N\times V}$ are latent variable estimates, i.e., kth set of N source estimates, which in our setting, correspond to the kth set of N feature estimates. The estimates of the feature span the low dimensional representation space and will be used to train a machine learning algorithm for the detection of misinformation. The estimates of $\mathbf{A}^{[k]}$ contribute to knowledge discovery during a high-impact event since they encode information about the connections between the high level feature vectors and the low dimensional representation space (Boukouvalas et al., 2020). It is worth noting that when $K = 1$, (1) reduces to a simple blind source separation (BSS) problem with one modality and the most popular way to achieve BSS is by using ICA (Comon & Jutten, 2010; Hyvärinen et al., 2004; Adalı et al., 2014).

As shown in Fig. 1, IVA provides a smart connection across multiple datasets by defining a source component vector (SCV), which enables one to take full statistical information across multi-modal datasets, enabling *true fusion* of multi-modal data. Using the random vector notation (as opposed to the one written using observations in 1), we write $\mathbf{x}^{[k]} = \mathbf{A}^{[k]}\mathbf{s}^{[k]}$, $k = 1, \ldots, K$, where $\mathbf{A}^{[k]} \in \mathbb{R}^{N\times N}$, $k = 1, \ldots, K$ are invertible mixing matrices and $\mathbf{s}^{[k]} = \left[s_1^{[k]}, \ldots, s_N^{[k]}\right]^\top$ is the vector of features for the kth dataset. In the IVA model, dependence across corresponding components of $\mathbf{s}^{[k]}$ is taken into account through the SCV which is obtained by vertically concatenating the nth source from each of the K dataset as $\mathbf{s}_n = \left[s_n^{[1]}, \ldots, s_n^{[K]}\right]^\top$ (Adalı et al., 2014). The goal in IVA is to estimate K demixing matrices to yield source estimates $\mathbf{y}^{[k]} = \mathbf{W}^{[k]}\mathbf{x}^{[k]}$, such that each SCV is maximally independent of all other SCVs. We note that while we consider the noiseless BSS model, the effect of noise is taken into account through dimension reduction such that we start with

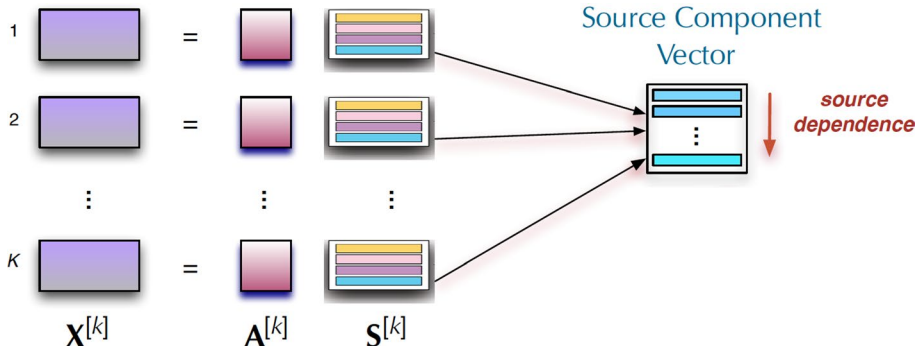

Fig. 1 IVA model for multi-modal data fusion and the statistical property taken into account: statistical dependence across modalities within a source component vector $\mathbf{s}_n$

overdetermined problems where $d > N$ and use a dimensionality reduction technique like principal component analysis (PCA) to project the data to a lower dimensional space where $d = N$. This simple step is critical for multi-modal data fusion since each modality might exhibit different levels of noise, and thus, identifying the optimal joint signal subspace would help improve the generalization abilities of the solution.

The IVA optimization parameter is defined as a set of demixing matrices $\mathbf{W}^{[1]}, \ldots, \mathbf{W}^{[K]}$, which can be collected into a three dimensional array $\mathcal{W} \in \mathbb{R}^{N\times N\times K}$ and can be estimated through the minimization of the IVA objective function given by

$$J_{IVA}(\mathcal{W}) = \sum_{n=1}^{N} H(\mathbf{y}_n) - \sum_{k=1}^{K} \log \left|\det\left(\mathbf{W}^{[k]}\right)\right| + C. \tag{2}$$

Here $H(\mathbf{y}_n)$ denotes the (differential)[1] entropy of the estimated nth SCV that serves as the term for modeling the complex relationships among the different modalities. By definition, the term $H(\mathbf{y}_n)$ can be written as $\sum_{k=1}^{K} H(y_n^k) - I(\mathbf{y}_n)$, where $I(\mathbf{y}_n)$ denotes the mutual information within the nth SCV. Therefore, it can be observed that minimization with respect to each demixing matrix $\mathbf{W}^{[k]}$ of (2) automatically increases the mutual information within the components of an SCV, revealing how IVA exploits statistical dependence across different modalities.

Using the IVA objective function (2), the derivative with respect to each of the demixing matrices is given by

$$\frac{\partial J_{IVA}(\mathcal{W})}{\partial \mathbf{W}^{[k]}} = E\left\{\boldsymbol{\phi}^{[k]}(\mathbf{x}^{[k]})^{\top}\right\} - (\mathbf{W}^{[k]})^{-\top}, \tag{3}$$

where $\boldsymbol{\phi}^{[k]} = -\left[\frac{\partial \log p_{s_1}(y_1)}{\partial y_1^{[k]}}, \ldots, \frac{\partial \log p_{s_N}(y_N)}{\partial y_N^{[k]}}\right]^{\top}$.

Thus, each of the K demixing matrices is updated using gradient descent

$$(\mathbf{W}^{[k]})^{\text{new}} \leftarrow (\mathbf{W}^{[k]})^{\text{old}} - \gamma \frac{\partial J_{IVA}(\mathcal{W})}{\partial \mathbf{W}^{[k]}}, \tag{4}$$

[1] We consider continuous-valued random variables and in the sequel, refer to differential entropy as simply entropy for simplicity.

where γ is the step size.

It can be observed that performing the optimization procedure on the space of all invertible matrices may result in poor convergence due to inversion of the $\mathbf{W}^{[k]}$ matrix at each iteration. A potential solution for this issue is to post multiply the gradient of the objective function by $(\mathbf{W}^{[k]})^\top(\mathbf{W}^{[k]})$. Although this "natural gradient" approach has shown significant results in terms of its convergence properties (Amari & Douglas, 1998; Cichocki & Yang, 1996), there are still limitations associated with optimization using matrix parameters. For instance, the term $E\{\boldsymbol{\phi}^{[k]}(\mathbf{x}^{[k]})^\top\}$ in (3) may be especially complicated when the class of estimated probability density functions (PDFs) for the estimated features is complicated. This motivates the division of the minimization of (2) into a series of sub-problems such that we minimize the objective function with respect to each of the row vectors $\mathbf{w}_1^{[k]}, \ldots, \mathbf{w}_N^{[k]}$ individually. This simplifies the density matching problem as the estimation of a given source will not affect the estimation of the others. In addition, optimization of cost functions with respect to each row vector of the demixing matrix *enables integration of flexible multivariate PDF estimation techniques* (see Sect. 2.1), *simplifies the incorporation of constraints in the IVA framework* (see Sect. 2.2), and *enables the implementation of parallel IVA algorithms* (see Sect. 3.1). Through this work, we will provide solutions to all of these aspects.

We mathematically formulate this by following Li and Adalı (2010), Anderson et al. (2012), Boukouvalas (2018). In the following discussion, we consider the cost function (2) and update rule (4) without the superscript $[k]$ to keep the notation simple. Let $\mathbf{W}_n = [\mathbf{w}_1, \ldots, \mathbf{w}_{n-1}, \mathbf{w}_{n+1}, \ldots, \mathbf{w}_N]^\top \in \mathbb{R}^{(N-1)\times N}$ denote the matrix that contains all rows of $\mathbf{W}$ except the nth one. Since the determinant of a matrix is invariant under row permutation up to a sign ambiguity, the square of the $\det(\mathbf{W})$ term in (2) is written as

$$\begin{aligned}\det(\mathbf{W})^2 &= \det(\mathbf{W}\mathbf{W}^\top) = \det\left(\begin{bmatrix}\mathbf{W}_n\\ \mathbf{w}_n^\top\end{bmatrix}\begin{bmatrix}\mathbf{W}_n\mathbf{w}_n^\top\end{bmatrix}\right) = \det\left(\begin{bmatrix}\mathbf{W}_n\mathbf{W}_n^\top & \mathbf{W}_n\mathbf{w}_n\\ \mathbf{w}_n^\top\mathbf{W}_n^\top & \mathbf{w}_n^\top\mathbf{w}_n\end{bmatrix}\right)\\ &= \det(\mathbf{W}_n\mathbf{W}_n^\top)\mathbf{w}_n^\top(\mathbf{I}-\mathbf{W}_n^\top(\mathbf{W}_n\mathbf{W}_n^\top)^{-1}\mathbf{W}_n)\mathbf{w}_n,\end{aligned} \tag{5}$$

where the term $\mathbf{H}_n = \mathbf{I} - \mathbf{W}_n^\top\left(\mathbf{W}_n\mathbf{W}_n^\top\right)^{-1}\mathbf{W}_n$ is the orthogonal projection onto the null space of $\mathbf{W}_n$. By definition, the matrix $\mathbf{H}_n$ is rank one, and thus, $\mathbf{H}_n = \mathbf{h}_n\mathbf{h}_n^\top$, where $\mathbf{h}_n$ is perpendicular to all row vectors of $\mathbf{W}_n$. Thus,

$$\begin{aligned}|\det(\mathbf{W})| &= \sqrt{\det(\mathbf{W}_n\mathbf{W}_n^\top)^2\mathbf{w}_n^\top\mathbf{h}_n\mathbf{h}_n^\top\mathbf{w}_n} = \sqrt{\det(\mathbf{W}_n\mathbf{W}_n^\top)^2(\mathbf{h}_n^\top\mathbf{w}_n)^2}\\ &= |\det(\mathbf{W}_n\mathbf{W}_n^\top)||(\mathbf{h}_n^\top\mathbf{w}_n)|.\end{aligned} \tag{6}$$

Therefore, by reintroducing the superscript $[k]$, the cost function (2) can be written as

$$J_{IVA}(\mathbf{w}_n^{[k]}) = \sum_{n=1}^{N} H(\mathbf{y}_n) - \log|(\mathbf{h}_n^\top\mathbf{w}_n^{[k]})| - \log|\det((\mathbf{W}_n^{[k]})(\mathbf{W}_n^{[k]})^\top)| - H(\mathbf{x}^{[k]}), \tag{7}$$

where the terms $H(\mathbf{x}^{[k]})$ and $\log|\det((\mathbf{W}_n^{[k]})(\mathbf{W}_n^{[k]})^\top)|$ are independent of $\mathbf{w}_n^{[k]}$. The gradient of (7) w.r.t. $\mathbf{w}_n^{[k]}$ is given by

$$\frac{\partial J_{\text{IVA}}}{\partial \mathbf{w}_n^{[k]}} = E\{\boldsymbol{\phi}_n^{[k]}(\mathbf{y}_n)\mathbf{x}^{[k]}\} - \frac{\mathbf{h}_n^{[k]}}{\left(\mathbf{h}_n^{[k]}\right)^\top\mathbf{w}_n^{[k]}}. \tag{8}$$

Thus, the estimate of each $\mathbf{W}^{[k]}$ can be determined w.r.t. each row vector $\mathbf{w}_n^{[k]}$, $n = 1, \ldots, N$ independently, by using the gradient update rule

$$(\mathbf{w}_n^{[k]})^{\text{new}} \leftarrow (\mathbf{w}_n^{[k]})^{\text{old}} - \gamma \frac{\partial J_{IVA}(\mathbf{w}_n^{[k]})}{\partial \mathbf{w}_n^{[k]}}. \tag{9}$$

A pseudocode description of an IVA algorithm is given in Algorithm 1 below. The main part of this algorithm is the loop described in lines 3-12, where one can observe that optimal convergence, and thus, true fusion of multi-modal data depends on *the development of effective models for the multivariate PDFs of each estimated SCV and their estimation as well as on efficient utilization of prior information through meaningful constraints*. In addition, one can observe that since the bulk of the computational complexity of IVA occurs in lines 3-12 in Algorithm 1, distributing separate iterations of the main loop to separate computational resources is desirable to reduce the total execution time.

Algorithm 1 IVA

1: **Input**: $\mathbf{X} \in \mathbb{R}^{N \times V \times K}$
2: For each $k = 1, .., K$, initialize $\mathbf{W} \in \mathbb{R}^{N \times N}$
3: **for** n = 1:N **do**
4: Estimate the multivariate PDF of the nth SCV, $\hat{p}(\mathbf{y}_n)$, to fully characterize $\phi_n^{[k]}(\mathbf{y}_n)$ in (8)
5: Compute $\mathbf{h}_n^{[k]}$ for $k = 1, ..., K$, which are orthogonal to $\mathbf{w}_i^{[k]}$ for all $i \neq n$
6: **for** k = 1:K **do**
7: Calculate the derivative $\frac{\partial J_{\text{IVA}}}{\partial \mathbf{w}_n^{[k]}}$
8: $(\mathbf{w}_n^{[k]})^{\text{new}} \leftarrow (\mathbf{w}_n^{[k]})^{\text{old}} - \gamma \frac{\partial J_{\text{IVA}}}{\partial \mathbf{w}_n^{[k]}}$
9: **end for**
10: **end** % k
11: **end for**
12: **end** % n
13: $J_{\text{IVA}} = \sum_{n=1}^{N} H(\mathbf{y}_n) - \sum_{k=1}^{K} \log \left| \det \left(\mathbf{W}^{[k]} \right) \right|$
14: Repeat steps 3 to 12 until convergence in $\mathscr{W}$ or maximum iterations exceeded
15: **Output**: $\mathscr{W}$

2.1 Effective density models for capturing multi-modal associations—IVA-M-EMK

The key factor in the explicit modeling of the non-linear relationships across different modalities is the estimation of the true underlying PDF of each estimated SCV. It is clear that minimizing (2) is not a straightforward task since there is no access to the true underlying PDF of each estimated SCV. To mathematically demonstrate this, if $\hat{p}(\mathbf{y}_n)$ denotes the PDF of the nth estimated SCV then its entropy can be expressed as

$$H(\mathbf{y}_n) = -f(p(\mathbf{y}_n), \hat{p}(\mathbf{y}_n)) - E\{\log \hat{p}(\mathbf{y}_n)\}, \tag{10}$$

where $f(p(\mathbf{y}_n), \hat{p}(\mathbf{y}_n))$ denotes the Kullback-Leibler (relative entropy) distance between the density of the nth estimated SCV and the true density of $\mathbf{y}_n$. From (10), we can achieve perfect source estimation as long as the assumed model PDF matches the true latent multivariate density of the nth SCV, i.e., $f(p(\mathbf{y}_n), \hat{p}(\mathbf{y}_n)) = 0$. As demonstrated in Boukouvalas et al.

(2018); Damasceno et al. (2021, 2022), PDF estimators based on the maximum entropy principle can successfully match multivariate latent sources from a wide range of distributions. The maximum entropy distribution for each $\mathbf{y}_n$ is given by

$$\hat{p}(\mathbf{y}_n) = \exp\left\{-1 + \sum_{m=0}^{M} \lambda_m r_m(\mathbf{y}_n)\right\}, \tag{11}$$

where the Lagrange multipliers λ_m are chosen such that the M number of moment constraints are satisfied.

Thus, the development of *flexible* and *efficient* models for entropy, their estimation using the maximum entropy principle, and their effective integration into (2), requires that we address the following three key issues:

1. Lagrangian multipliers evaluation and choice of constraints:

We evaluate the Lagrangian multipliers by the Newton iteration scheme using local and global constraints. The estimation of the Lagrange multipliers highly depends on the proper selection of the constraints in order to provide information about the underlying statistical properties of the data. Failing on this will result in high complexity and poor data characterization.

Following a similar strategy as in Fu et al. (2015), Damasceno et al. (2021, 2022), we jointly use global and local constraints to provide flexible multivariate density estimation while keeping the complexity low. Therefore, we use $\mathbf{1}, \mathbf{y}_n, \mathbf{y}_n^2, \mathbf{y}_n/(\mathbf{1}+\mathbf{y}_n^2)$ as the global constraints, since they provide information on the PDF's overall statistics, such as the mean, variance, and higher order statistics (HOS). For the local constraint we use the Gaussian kernel given by,

$$q(\mathbf{y}_n) = \frac{1}{\sqrt{|\boldsymbol{\Sigma}_n|(2\pi)^K}} \exp\left(-\frac{1}{2}(\mathbf{y}_n - \boldsymbol{\mu}_n)^\top \boldsymbol{\Sigma}_n^{-1}(\mathbf{y}_n - \boldsymbol{\mu}_n)\right), \tag{12}$$

where $\boldsymbol{\mu}_n$ denotes the mean vector, $\boldsymbol{\Sigma}_n$ denotes the covariance matrix, and $|\cdot|$ denotes the determinant. The Gaussian kernel provides localized information about the PDF.

It is important to mention that when we add the Gaussian kernel to the multi-dimensional framework, integration becomes challenging because the Gaussian kernel has an infinite support.

2. Multi-dimensional integration during the estimation of the Lagrange multipliers:

Multi-dimensional integration is one of the main challenges in our estimation problem. We use an efficient multi-dimensional integration technique based on Quasi-Monte Carlo methods (QMC) to overcome this problem. QMC have shown to be efficient in terms of their rate of convergence and achieve a convergence rate of order $O((\log V)^K/V)$ (Dick et al., 2013). Following the steps in Damasceno et al. (2021), we generate a sequence of quasi-random points (Niederreiter, 1992). Using this sequence, we approximate the multi-dimensional integrals in a Monte Carlo method manner (Dick et al., 2013).

3. Efficient multivariate density estimation technique based on IVA by multivariate entropy maximization with kernels (IVA-M-EMK):

Once the Lagrange multipliers have been estimated, and we have a full characterization of the underlying PDF and, therefore, a full characterization of the entropy for each estimated SCV, IVA-M-EMK provides estimates of the demixing matrices by minimizing (2). The gradient of (2) with respect to each row vector $\mathbf{w}_n^{[k]}$ of $\mathbf{W}^{[k]}$ for the IVA-M-EMK is given by

$$\frac{\partial J_{\text{IVA}}}{\partial \mathbf{w}_n^{[k]}} = -\sum_{i=0}^{M} \lambda_i \frac{\partial r_i(\mathbf{y}_n)}{\partial \mathbf{y}_n^{[k]}} E\{\mathbf{x}^{[k]}\} - \frac{\mathbf{h}_n^{[k]}}{\left(\mathbf{h}_n^{[k]}\right)^{\top} \mathbf{w}_n^{[k]}}. \tag{13}$$

Following the idea in Boukouvalas et al. (2018), we perform the optimization routine in a Riemannian manifold rather than a classical Euclidean space since this provides important convergence advantages. We define the domain of our cost function to be the unit sphere in $\mathbb{R}^N$ and project (13) onto the tangent hyperplane of the unit sphere at the point $\mathbf{w}_n^{[k]}$. Since the IVA-M-EMK cost function depends on the number of moment constraints chosen for each SCV, non-monotonic behavior is expected between two consecutive iterations.

2.2 Sparsity constraints to capture conditional independence between estimated joint features—IVA-SPICE

IVA relies on the assumption of statistical independence of the latent features. Although this might be a natural assumption in many problems, it may be too strong for our application. Incorporating prior reliable and meaningful information about the underlying multimodal features can help relax the statistical independence assumption, resulting in a better model match. This will result in better estimation of the SCVs, and thus, its corresponding estimated precision matrices will better reveal associations between extracted low dimensional features across the modalities. A logical approach to isolate the most important relationships of the underlying joint features is to impose a Gaussian model on each SCV parameterized by the inverse covariance matrix to induce sparsity.

Thus, under the assumption that each SCV follows a multivariate Gaussian distribution we have that

$$\phi_n^{[k]}(\mathbf{y}_n) = \mathbf{y}_n^{\top} \Sigma_n^{-1} \mathbf{e}_k, \tag{14}$$

where $\mathbf{e}_k$ is the unit vector. Therefore, the gradient of (2) with respect to each row vector $\mathbf{w}_n^{[k]}$ of $\mathbf{W}^{[k]}$ is given by

$$\frac{\partial J_{\text{IVA}}}{\partial \mathbf{w}_n^{[k]}} = E\{\mathbf{x}^{[k]} \mathbf{y}_n^{\top}\} \Sigma_n^{-1} \mathbf{e}_k - \frac{\mathbf{h}_n^{[k]}}{\left(\mathbf{h}_n^{[k]}\right)^{\top} \mathbf{w}_n^{[k]}}. \tag{15}$$

Taking advantage of the natural properties of the precision matrix provides a known structure which increases potential use cases for IVA. Since, under the Gaussian assumption, we can leverage the equivalence of partial correlation and conditional independence among the joint features, sparsity of the precision matrix is an informative property. Leveraging sparsity in the precision matrix reduces the effects of confounding joint features, thereby capturing conditional dependencies in the underlying multi-modal data structure. Such conditional dependencies appear as zero values in the inverse covariance (Dempster, 1972; Lauritzen, 1996). Moreover, exploitation of the sparse structure of each SCV reduces the

number of parameters to be estimated when we have assumed that the SCVs follow a multivariate Gaussian distribution. However, each precision matrix Σ_n^{-1} is most often unknown; therefore, its efficient estimation plays a crucial role in the overall performance of IVA.

Graphical Lasso for the estimation of each Σ_n^{-1}. To simplify the notation we drop the under-script n from Σ^{-1}. Let us consider the Gaussian log-likelihood function $\ell\left(\boldsymbol{\Sigma}^{-1}\right) = \mathrm{tr}\left(\boldsymbol{S}\boldsymbol{\Sigma}^{-1}\right) - \mathrm{logdet}\boldsymbol{\Sigma}^{-1}$, where $\boldsymbol{S}$ is the empirical covariance. Lasso penalization of Tibshirani (1996) based on the $\ell 1$ norm – the sum of absolute values of entries in Σ^{-1} – is a popular approach to estimate the precision matrix. The graphical lasso algorithm presented in Friedman et al. (2007) yields a sparse inverse covariance, effectively isolating the strongest relationships between variables in the data by imposing a specified degree of sparsity.

Mathematically, graphical lasso solves the convex optimization problem

$$\hat{\boldsymbol{\Sigma}}^{-1} = \mathrm{argmax}_{\Sigma^{-1}} \mathrm{tr}\left(\boldsymbol{S}\boldsymbol{\Sigma}^{-1}\right) - \mathrm{logdet}\boldsymbol{\Sigma}^{-1} - \rho\|\boldsymbol{\Sigma}^{-1}\|_1, \tag{16}$$

where the scalar parameter ρ controls the magnitude of the penalty on the $\ell 1$ norm, $\|\boldsymbol{\Sigma}^{-1}\|_1$. The choice of ρ decides a trade off between the maximum likelihood (ML) and sparsity; smaller values fit the data better, but larger values encourage sparser solutions. A popular choice for ρ is the cross-validated choice, which typically improves model performance.

Following the work in Banerjee et al. (2008), the optimization problem in (16) is solved by estimating $\boldsymbol{\Sigma}$ as $\boldsymbol{M}$ through a blockwise updating scheme. This involves optimizing over each row and corresponding column of $\boldsymbol{M}$. $\boldsymbol{M}$ and $\boldsymbol{S}$ are partitioned as

$$\boldsymbol{M} = \begin{pmatrix} M_{11} & m_{12} \\ m_{12}^{\mathrm{T}} & m_{22} \end{pmatrix}, \boldsymbol{S} = \begin{pmatrix} S_{11} & s_{12} \\ s_{12}^{\mathrm{T}} & s_{22} \end{pmatrix}. \tag{17}$$

In Friedman et al. (2007), blockwise coordinate descent is used for the estimation of $\boldsymbol{\Sigma}^{-1}$ in graphical lasso, letting $\boldsymbol{M}\boldsymbol{\Sigma}^{-1} = \boldsymbol{I}$. They show the subgradient for the maximization of (16) is

$$\boldsymbol{M} - \boldsymbol{S} - \rho\boldsymbol{\Gamma} = 0, \tag{18}$$

where $\boldsymbol{\Gamma}$ is determined by the signs of $\boldsymbol{\Sigma}$:

$$\gamma_{jk} = sign(\Sigma_{jk}^{-1}) \text{ if } \Sigma_{jk}^{-1} \neq 0, \quad \gamma_{jk} \in [-1, 1] \text{ if } \Sigma_{jk}^{-1} = 0. \tag{19}$$

and is partitioned the same way as (17). As per the coordinate-wise pattern, graphical lasso solves for one row/column at a time while holding the rest fixed. The Kth column of (18) gives

$$-m_{12} + s_{12} + \rho\gamma_{12} = 0. \tag{20}$$

They show that m_{12} can be accessed as:

$$m_{12} = -M_{11}\Sigma_{12}^{-1}/\Sigma_{22}^{-1}, \tag{21}$$

and plugging this into (20) gives the gradient equation,

$$M_{11}\frac{\Sigma_{12}^{-1}}{\Sigma_{22}^{-1}} + s_{12} + \rho\gamma_{12} = 0, \tag{22}$$

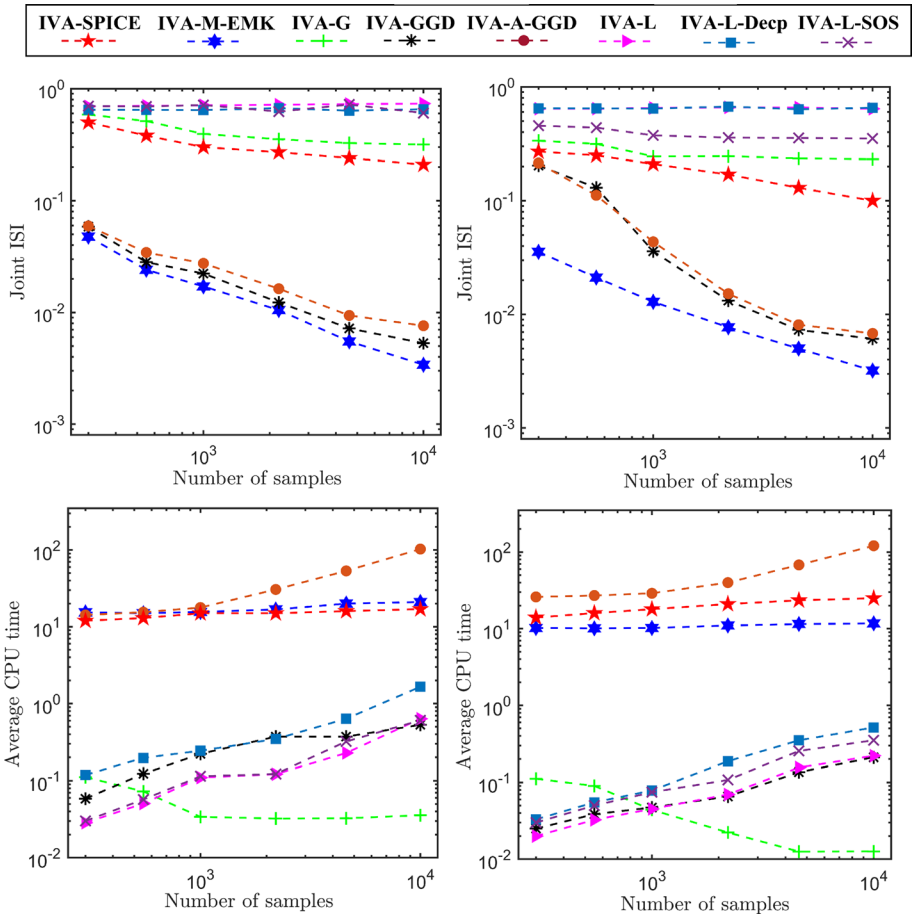

Fig. 2 Performance comparison in terms of Joint ISI and average CPU time for different number of sample sizes. The first and second case represented in the first and second columns, respectively. For the first case, we have $K = 3$ and generate one unimodal MGGD SCV where the shape parameter and the correlation within the SCV for each dataset are chosen to be $\beta = 3$ and $\mu = 0.6$, and a mixture of two MGGD sources where $\beta \in (0.6, 0.8)$ and $\mu \in (5, 10)$ respectively. For the second case, we have $K = 2$ and generate three SCVs where each SCV is a mixture of MGGD sources where β, μ are chosen from the range (0.5, 1) and (0.5, 10) respectively. Average CPU time is measured in seconds

which graphical lasso solves for with $\beta = \Sigma_{12}^{-1} / \Sigma_{22}^{-1}$. Finally, they define the lasso problem to be solved coordinate-wise as:

$$\min_{\beta} \left\{ \frac{1}{2} \beta^{\mathsf{T}} M_{11} \beta + \beta^{\mathsf{T}} s_{12} + \rho \|\beta\|_1 \right\}, \tag{23}$$

It is clear that $\hat{m}_{12}$ is found from β in (21). $\hat{\Sigma^{-1}}_{22}$ can be found by:

$$\frac{1}{\hat{\Sigma}_{22}^{-1}} = m_{22} - \hat{\beta}^{\mathsf{T}} \hat{m}_{12}. \tag{24}$$

And now, it is easy to estimate $\hat{\Sigma}_{12}^{-1}$ from $\hat{\beta}$ and $\hat{\Sigma}_{22}^{-1}$.

Given that graphical lasso estimates $\hat{\boldsymbol{\Sigma}}^{-1}$ regardless of the relationship between K and V, it avoids the constraints faced by empirical estimation of $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{n}}{}^{-1}$. Therefore, graphical lasso is an ideal solution for the development of a new algorithm: IVA with sparse inverse covariance estimation, or IVA-SPICE. After each precision matrix has been estimated and we have a full characterization of the underlying multivariate Gaussian parameterized by a sparse precision matrix, IVA-SPICE provides estimates through an optimization strategy similar to the one described in the IVA-M-EMK case.

3 Results based on simulated data

For the first set of our experiments, we show the effectiveness of the IVA-M-EMK and IVA-SPICE algorithms[2] by comparing their performance with six widely used IVA algorithms in terms of CPU time and terms of the joint inter-symbol-interference (ISI) as defined in Anderson et al. (2012). Joint ISI is a global metric for evaluating the separation performance when the ground truth $\left(\text{each mixing matrix } \mathbf{A}^{[k]}\right)$ is available. Here, a zero ISI indicates perfect separation, while a Joint ISI equal to one indicates the worst separation. For the following experiments, we consider two cases when generating the data for the SCVs. For the first case, we have $K = 3$ and generate one unimodal multivariate generalized Gaussian distribution (MGGD) SCV where the shape parameter and the correlation within the SCV for each dataset are chosen to be $\beta = 3$ and $\mu = 0.6$, and a mixture of two MGGD sources where $\beta \in (0.6, 0.8)$ and $\mu \in (5, 10)$ respectively. For the second case, we have $K = 2$ and generate three SCVs where each SCV is a mixture of MGGD sources where β, μ are chosen from the range (0.5, 1) and (0.5, 10) respectively. Results are averages of 30 runs.

From Fig. 2, we see that IVA-generalized Gaussian distribution (IVA-GGD) and IVA-adaptive generalized Gaussian distribution (IVA-A-GGD) provide a desirable performance for the first case in terms of Joint ISI as a function of sample size, revealing the flexibility of their underlying density models. Conversely, the algorithms based on IVA-Laplacian (IVA-L) (Kim et al., 2006) and IVA-Gaussian (IVA-G) (Anderson et al., 2012) do not provide a desirable performance due to assuming Laplacian and Gaussian distribution for the underlying sources. Overall, IVA-M-EMK performs the best among the eight algorithms due to its ability to successfully match multivariate latent sources from a wide range of distributions. IVA-SPICE does not provide high accuracy due to the model mismatch; however, it does a better job than IVA-G. In addition to assessing the IVA algorithms based on the mean Joint-ISI, we have examined the standard deviation of the Joint ISIs across various runs. An intriguing observation emerged: for $V = 10,000$, IVA-M-EMK demonstrated the most favorable mean Joint ISI, and its stability across multiple runs was attributed to the model match, which further led to the lowest standard deviation.

In terms of CPU time, among the algorithms that use a simple underlying density model, IVA-G provides the best performance for both cases. This is due to the assumption of Gaussian distribution for the underlying sources, simplifying the gradient of the IVA objective function, thus improving the quality of convergence. On the other hand, as we expect, IVA-M-EMK, IVA-A-GGD (Boukouvalas et al., 2015), and IVA-SPICE are more computationally expensive; however, for the IVA-M-EMK case, we see that as the number of samples increases, the increase in average CPU time is negligible.

[2] The code for IVA-M-EMK and IVA-SPICE are available at https://zoisboukouvalas.github.io/Code.html.

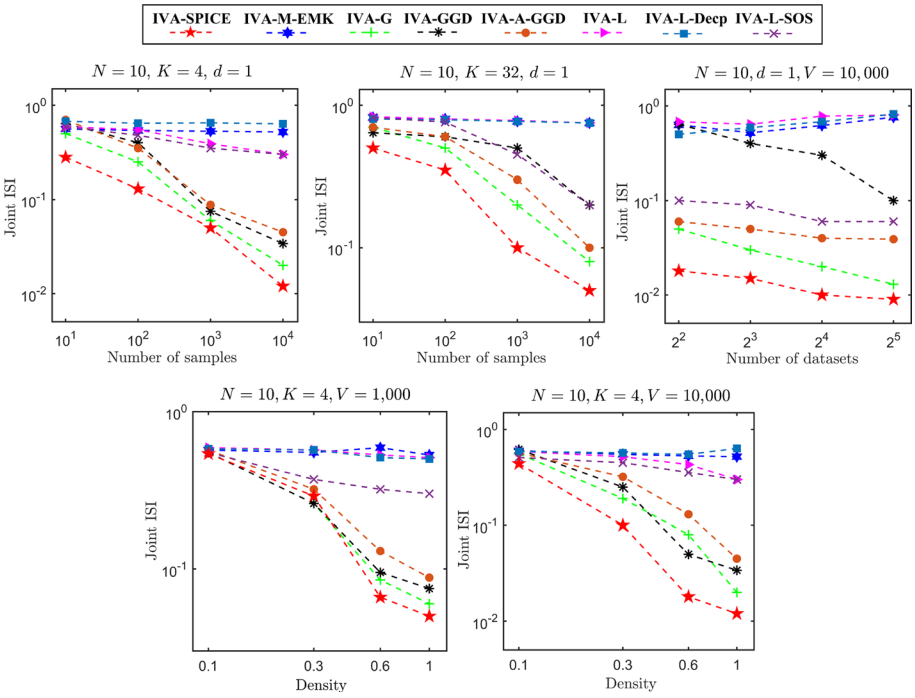

Fig. 3 Performance comparison in terms of Joint ISI for different number of sample size, different number of datasets (K), as well as different values of density of nonzero values on the true precision matrices. Average CPU time is measured in seconds

For the second set of experiments, we evaluate the performance of IVA-SPICE with simulated sparse input data. For all experiments, we define the nth SCV as a zero mean, K dimensional, multivariate Gaussian random vector parameterized by a sparse inverse covariance matrix $\boldsymbol{\Sigma}_n^{-1}$, where sparsity is measured by the density of nonzero values. We experiment with varying density d, which represents the sparsity of the generated data, whe, as well as the number of components N, datasets K, and samples V, with Joint ISI, averaged for ten trials.

Starting from the first row in Fig. 3, for the plots in the first and second column, we evaluate the source separation performance varying the number of samples from 100 to 10, 000 with $N = 10$, $K = 4$, $d = 1$ and $N = 10$, $K = 32$, $d = 1$, respectively. For this scenario, holding the density of nonzero values at 1, we can observe that in the sparser case, IVA-SPICE achieves the best separation performance. Moreover, from the right plot in the first line with $N = 10$, $d = 1$, and $V = 10,000$, we can see that increasing the number of datasets is negligible. Finally, for the plots in the second row, we fix $K = 4$ and once again hold $N = 10$ for $V = 1000$ in the first columns and $V = 10,000$ in the second. We can see that IVA-SPICE outperforms all the other IVA algorithms when there is underlying sparsity in the SCVs. Furthermore, an important aspect is that IVA-SPICE greatly improves its source separation performance with a large rate of decrease when more samples are available. Increasing the sample size in our experiments represents the optimal solution since IVA is formulated under a maximum likelihood framework (Damasceno et al., 2021). Last, we have investigated the standard deviations across multiple runs while varying two

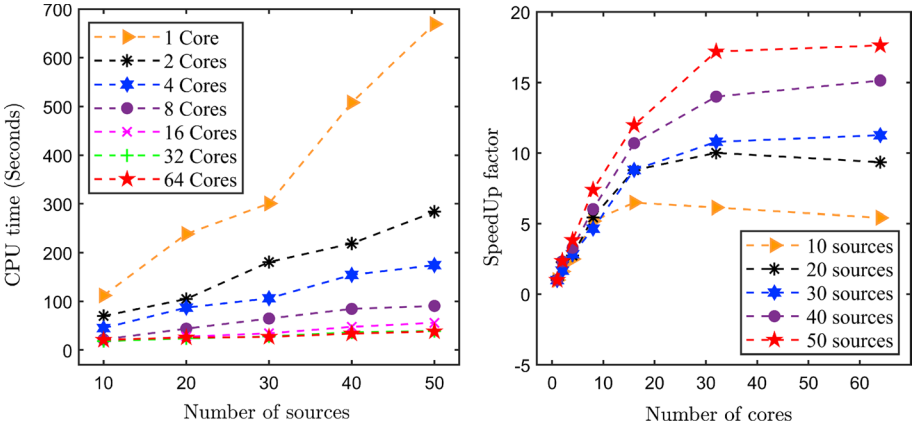

Fig. 4 Scalability experiments for the parallel SPICE-IVA implementation in terms of CPU time (left) and SpeedUp factor (right). SpeedUp is computed as the ratio between the time execution with the single-core implementation, and the corresponding execution time with multiple cores

parameters: K (Fig. 3, row 1, column 3) and the density (Fig. 3, row 2, column 1). As we increase K, algorithms with more flexible underlying models, such as IVA-M-EMK and IVA-SPICE, exhibit higher standard deviations across multiple runs. However, IVA-M-EMK fails to deliver a desirable mean Joint ISI due to model mismatch and underlying sparsity issues. On the other hand, when increasing the density, IVA-SPICE, as expected, provides the lowest mean Joint ISI. Yet, its model flexibility (Graphical Lasso) causes higher standard deviation compared to the other IVA algorithms. In particular, the standard deviation of IVA-SPICE when $d = 1$ is 0.193 and for the same case the standard deviation of IVA-SPICE is 0.125.

3.1 Parallel implementation of IVA

A highly-regarded characteristic of machine learning tools is the ability to support the analysis of large-scale data, which is frequently encountered in real-world applications. Since the bulk of the computational complexity of an IVA algorithm mostly occurs during the estimation of the SCV PDFs, distributing separate iterations of the main loop to separate computation resources is desirable to reduce the total execution time. The decoupling trick, as presented in Sect. 2, transforms the matrix optimization task for (2) into a series of vector optimization problems and, at the same time, it provides independence between the computation of each of the IVA cost function gradient directions. Our optimized IVA implementation splits the input data across features, such that each task of estimating the PDF of each SCV is assigned to a separate CPU processor. In the final computation, the partial contribution from each estimated SCV is aggregated to obtain the final result. Our implementation leverages the Python multiprocessing API,[3] which allows us to control and assign workloads to multiple processors on a single computational node. As we observe from Fig. 2, IVA-SPICE is computationally expensive and since this algorithm is implemented in Python, we choose to demonstrate the computational speedup of the parallel IVA-SPICE implementation over its sequential counterpart.

[3] https://docs.python.org/3/library/multiprocessing.html.

Experiments in Fig. 4 show the CPU time (left) and the SpeedUp factor (right) obtained with an increasing number of sources and cores, respectively. Experiments were performed on a workstation equipped with an Intel Xeon Gold CPU with 52 cores (104 threads) and 256GB RAM. The results highlight that a significant speedup can be obtained with the parallel implementation of IVA-SPICE as more sources and CPU cores are available. Focusing on experiments with $N = 50$, it can be observed that the execution with 64 cores brings down the execution time from 669.18 to 37.98 s, which results in a speedup of 17.62x. Even though this result is far from the theoretical upper bound (64x), the reduction margin achieved in the execution time is remarkable. Generally, the results show that increasing the number of cores always provides an improvement in terms of speedup and a consequent reduction in the execution time. It is worth noting that, in our experiments, no differences in the ISI results were observed for the single-core vs. multi-core implementations. As a result, the reduction in the execution time achieved with the parallel implementation does not come at the cost of reduced accuracy for the IVA-SPICE converging solution. Future work includes further optimization of the method to provide linear scalability on cluster computing environments, exploiting distributed CPU and GPU-based programming frameworks, as well as the parallel implementations of most IVA algorithms and, in particular, IVA-M-EMK.

4 Application to misinformation detection

4.1 Dataset

We utilized the final processed training and test datasets from our previous study (Damasceno et al., 2022). These datasets were created using the MediaEval2016 Image Verification Corpus[4] (Boididou et al., 2018). The datasets include:

1. Separately labeled training and test tweet text and multimedia datasets from the 2016 vintage.
2. The training dataset comprises tweets revolving around a set of events different from those in the test dataset.
3. Each tweet is labeled as “fake” or “real," where “fake" refers to multimedia content that does not faithfully represent the event it refers to.
4. “Fake" content includes posts with past event media misrepresented as currently unfolding, manipulated context, or false claims about the depicted event.

The working datasets include records with a single corresponding image. The text data consists of "clean" text without emoji characters, stop words, URLs, Twitter handles, time stamps, select punctuation, and words less than two characters. The text data was normalized by lowercasing each word, reducing multi-spaces to one space, and lemmatizing the text. The dataset includes only tweets identified as using English or a similar language using the Langid Python package and the International Organization for Standardization (ISO) code for languages. The text data does not include tweets that were denoted as being retweeted. Records that resulted in null values during prior feature extraction are not

[4] https://github.com/MKLab-ITI/image-verification-corpus.

present. The image data was previously pre-processed by resizing the images to 224 × 224 pixels and normalizing them.

The training dataset consists of:

- 9140 tweet records associated with 352 different images.
- Representing 15 unique events.
- 5127 tweets labeled as fake.
- 4013 tweets labeled as real.
- Five events include both real and fake tweets.
- Ten events include only fake tweets.

The test dataset consists of:

- 796 tweet records associated with 92 different images.
- Representing 23 unique events.
- 467 tweets labeled as fake.
- 329 tweets labeled as real.
- Seven events have both real and fake tweets.
- One event includes only real tweets.
- Fifteen events include only fake tweets.

We also include features extracted during the previous study in the starting datasets. As a text feature, we include a 300-dimensional Word2Vec (Mikolov et al., 2013) embedding vector for each tweet record where each tweet is represented by the average word embedding vectors of the words that make up the tweet. This feature was created using a Word2Vec model trained on the Google News corpus.[5] As a high-level image feature, we include a 4,096-dimensional fully connected layer from a pre-trained VGG-16 model for each image associated with an individual tweet record, as these yielded the best classification results during model evaluations in the previous study. It is worth mentioning that we evaluated features generated using Bidirectional Encoder Representations from Transformers, known as BERT (Devlin et al., 2018). However, the classification accuracy for this specific dataset was not as high as the accuracy achieved using Word2Vec-based embedding vectors. As an illustration, the fusion of BERT-based embedding vectors with VGG-based image features yields an F1-score of 70.68%. On the other hand, when Word2Vec is combined with VGG, the F1-score increases to 77.45% as shown in Fig. 1b. It is important to mention that BERT comes in various versions, and exploring all these variations in terms of the F1-score would be beyond the scope of our current research work. However, it could be an interesting direction for future research.

4.2 Additional high level feature extraction

To assess the impact of additional features and modalities on the classification accuracy and performance of the algorithms, we create two new features to include in the analysis. The first additional feature we create we call "Image2text". To make this feature, we use an Image Captioning PyTorch model[6] pre-trained with ResNet101 features to generate

[5] https://code.google.com/archive/p/word2vec/

[6] https://github.com/ruotianluo/ImageCaptioning.pytorch.

captions for each image in our datasets. We then use the same Word2Vec model we previously used to create a 300-dimensional Word2Vec embedding using the average word embedding vectors for each word in each generated caption. This provides an additional text feature that we use as a separate modality for our evaluations. The second feature we create and utilize as another modality for our analysis is a 200-dimension vector representing the top 200 topics assigned to a tweet using topic modeling. To conduct the topic modeling, first we utilize the term frequency-inverse document frequency (TF-IDF) vectorizer from the scikit-learn Python package to generate a TF-IDF matrix for the text from the tweets, using a vocabulary that consists of words that are in less than 95 percent and in more than one percent of the tweets in the datasets. We then apply non-negative matrix factorization (NMF) to assign 200 topics to each tweet. We use 200 topics to ensure that the feature's dimensions are compatible with our other matrices to conduct our analysis.

4.3 Classification procedure

The classification process consists of four stages. As mentioned in Sect. 4.1, our dataset is separated into training and testing datasets, where each tweet is represented by text as well as visual content. With this in mind, in the first stage we form our set of tweets in the following way. We denote the training observation matrices for each modality with $\mathbf{X}_{\text{train}}^{[k]} \in \mathbb{R}^{d_k \times V_{\text{train}}}$ where d_k denotes the number of initial high-level feature vectors in each modality and V_{train} denotes the number of training tweets. Similarly, $\mathbf{X}_{\text{test}}^{[k]} \in \mathbb{R}^{d_k \times V_{\text{test}}}$ denotes the corresponding testing observation matrices. In the second stage, the mean from each dataset is removed so they are centered and PCA is applied to each $\mathbf{X}_{\text{train}}^{[k]}$, for $k = 1, 2, \ldots, K$. For the PCA step, we use an order N, which, in our setting, denotes the number of features from each modality. Then, for each $k = 1, 2, \ldots, K$, we obtain $\hat{\mathbf{X}}_{\text{train}}^{[k]} \in \mathbb{R}^{N \times V_{\text{test}}}$ and vertically concatenate each $\hat{\mathbf{X}}_{\text{train}}^{[k]}$ to form a three dimensional array $\hat{\mathbf{X}}_{\text{train}} \in \mathbb{R}^{N \times V_{\text{train}} \times k}$. In the third stage, we perform IVA on $\hat{\mathbf{X}}_{\text{train}}$, and since we have K modalities, IVA provides K demixing matrices $\mathbf{W}^{[k]} \in \mathbb{R}^{N \times N}$, for $k = 1, 2, \ldots, K$. Then, using the estimated demixing matrices we generate $\mathbf{Y}_{\text{train}}^{[k]} = \mathbf{W}^{[k]} \left(\hat{\mathbf{X}}_{\text{train}}^{[k]} \right)^{\top}$, for $k = 1, 2, \ldots, K$. The training dataset $\mathbf{Y}_{\text{train}}$ is formed by either concatenating, averaging, or max pooling the estimated SCVs which can be obtained by concatenating the estimated sources from $\mathbf{Y}_{\text{train}}^{[k]}$, for $k = 1, 2, \ldots, K$. Note that $\mathbf{Y}_{\text{train}}$ contains all the extracted features from the multi-modal data and it will be used for training the classification model. The testing dataset is generated by removing the training mean from each multi-modal testing dataset and using the generated PCA transformations from the training phase. The demixing matrices from the training phase are used to transform the testing datasets as follows, $\mathbf{Y}_{\text{test}}^{[k]} = \mathbf{W}^{[k]} \left(\hat{\mathbf{X}}_{\text{test}}^{[k]} \right)^{\top}$, for $k = 1, 2, \ldots, K$, where $\mathbf{Y}_{\text{test}}^{[k]} \in \mathbb{R}^{N \times V_{\text{test}}}$. Finally, the testing dataset $\mathbf{Y}_{\text{test}}$ is formed by either concatenating, averaging, or max pooling the estimated SCVs which can be obtained by concatenating the estimated sources from $\mathbf{Y}_{\text{test}}^{[k]}$, for $k = 1, 2, \ldots, K$. In the fourth stage, we train the classification model using $\left(\mathbf{Y}_{\text{train}} \right)^{\top}$. As mentioned in the introduction, the specific form of the classification model is unimportant. However, to demonstrate a concrete example, we use support vector machines (SVMs), which have shown reliable performance in a variety of applications, especially with smaller size datasets (Cortes & Vapnik, 1995; Moroney et al., 2021). Once the classification model has been trained, we evaluate its performance using the unseen dataset, $\left(\mathbf{Y}_{\text{test}} \right)^{\top}$. For all

experiments, hyper-parameter optimization and model training and testing is done using a grid search cross-validation with five folds scheme. The entire process was repeated five times (with shuffling before each iteration) to generate well converged statistics.

4.4 Classification performance using textual and visual high level features

For all of the experiments, we measure classification performance by employing the F1-score and reporting its macro averaged version. Moreover, we report the total CPU time of the training and testing phases and measure it in seconds. Before we introduce multiple modalities to show the power of IVA, we demonstrate the importance of combining multimodal datasets to improve the learned response of the classification model over the classification performance of the individual feature vectors treated separately. Thus, we compare the classification performance of three different classification models; one trained using only the high-level textual features, one trained using the high-level visual features, and one trained using the high-level textual features and the high-level visual features concatenated together.

From Table 1a, we see that if we train a classifier with just the high-level textual features, we obtain a classification performance of 40.04%, while if we train just using the high-level image features, we obtain an F1-score of 65.78%. If we concatenate the high-level text and image features, we obtain a classification performance of 77.59%. Note that the classification benchmark for the given F1-scores is 50%. This benchmark serves as a reference point to gauge the classification performance, and any F1-score above 50% indicates better than random classification. This result demonstrates that training a classifier using both modalities yields better classification performance. However, such an approach comes with significant challenges. As we can observe from Table 1a, concatenating the two modalities results in feature vectors of dimension 4396, thus affecting the efficiency of the machine learning algorithm. In addition, without exploiting the complementary information among multiple modalities, discovering the features of greatest importance and how they interact with each other becomes impossible.

Table 1 Classification performance in terms of F1-score for different classification scenarios

Methods	F1	CPU time (s)
(a) Regular approach		
Concatenate	**77.59%**	$\mathbf{1.7 \times 10^4}$
Text	40.04%	2.7×10^3
Image	65.78%	1.03×10^4
(b) IVA-M-EMK		
Concatenate	73.77%	2.4×10^3
Maximum	73.11%	1.5×10^3
Average	**77.45%**	$\mathbf{1.3 \times 10^3}$

Table 1a utilizes the concatenation method, involving the vertical concatenation of high-level features, such as text and images. In contrast, Table 1b employs the concatenation of the estimated SCVs from IVA, also arranged vertically. Total CPU execution time is measured in seconds

Bold values denote the highest level of performance

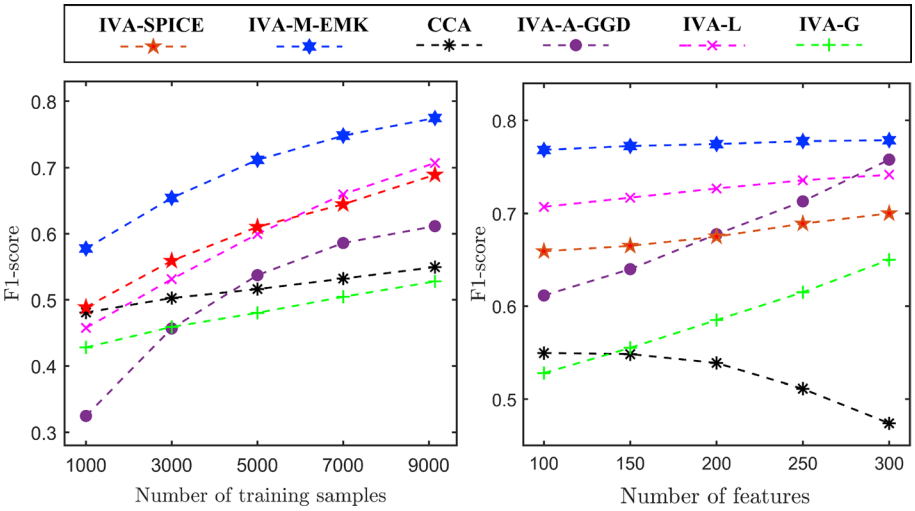

Fig. 5 Performance comparison in terms of F1-score and average CPU time for different number of features when all training samples are used

Due to its superior flexibility, IVA-M-EMK can address both challenges since it enables simultaneous study of multiple modalities by explicitly exploiting alignments of data fragments where there is a common underlying feature space. This can been seen from Table 1b, where IVA-M-EMK with $N = 100$ and averaging all SCVs leads to high classification accuracy and superior improvement in terms of the CPU execution time. Moreover, Table 1b shows two additional methods to combine the estimated SCVs after IVA-M-EMK has been applied with $N = 100$. Due to the fact, that the "Average" method yields the highest F1-score and lowest CPU execution time, for the rest of our experiments we adopt the "Average" method in order to combine the estimated SCVs. Numerical experiments showed that is true for all IVA algorithms.

For the following set of experiments, we compare IVA-M-EMK and IVA-SPICE with several IVA algorithms and canonical correlation analysis (CCA) in terms of the F1-score as a function of the number of features and training samples and when $K = 2$. It is worth mentioning that CCA does not explicitly impose an underlying density model for the joint features, but it implicitly seeks for a pair of vectors with the maximum correlation coefficient. On the other hand, different IVA algorithms *explicitly* model the underlying associations by assuming different probability densities for the underlying SCVs. In particular, IVA-L (Kim et al., 2006) is an algorithm that takes HOS into account and assumes a Laplacian distribution for the underlying SCVs. IVA-G (Anderson et al., 2012) exploits linear dependencies but does not take HOS into account. Finally, IVA-A-GGD (Boukouvalas et al., 2015) is a more general IVA implementation where both second and HOS are taken into account. This algorithm assumes an MGGD for the underlying SCVs, and through the estimation of its parameters, multivariate Gaussian and Laplacian distributions become special cases.

From the right plot in Fig. 5, we see that the F1-score for most IVA algorithms is invariant to the increase in features when all training samples are used. In addition, we see that as the number of features increases, IVA-M-EMK, IVA-A-GGD, IVA-SPICE, and IVA-L provide a desirable performance, followed by IVA-G. Conversely, as the

number of features increases, CCA provides the worst performance due to its model simplicity.

From the left plot in Fig. 5, we see that for $N = 100$ all IVA algorithms except IVA-G and CCA provide a desirable performance in terms of F1-score as the number of training samples increases. This reveals the flexibility of their underlying density models. A high F1-score as a function of training tweets makes IVA-M-EMK an ideal fusion approach for misinformation detection during high-impact events when textual and visual modalities are used. In addition, we see that IVA-SPICE provides good performance when less than 5,000 training samples are available, showing how important it becomes to reduce the effects of confounding joint features in cases with a poor training sample. Finally, we note that the classification results of our approach are on par with results obtained in similar studies such as Boididou et al. (2018).

Overall, IVA-M-EMK performs the best among the six algorithms. This can be attributed to its capability to explicitly model the underlying associations in the joint features by assuming different probability densities for the underlying SCV. The estimation of these probability densities is carried out using the maximum entropy principle, which has proven to be a highly effective method for multivariate probability density estimation (Damasceno et al., 2021).

4.5 Experimenting with more modalities

The proposed data fusion approach allows the quantification of each modality's additive value and the determination of the ideal combination of modalities that provides the highest prediction score. To demonstrate this, we evaluate the performance of IVA-M-EMK and IVA-SPICE as a function of the number of fused modalities. Table 2 displays the F1-scores as a function of the number of multi-modal data used to train the IVA transformations and classification model. For this experiment, we still choose $N = 100$ and average all estimated SCVs to avoid potential over-fitting. For comparison, when the classification algorithm is trained using only high-level Image2text features, the F1-score is 52.15%. When the classification algorithm is trained using only high-level topic features, the F1-score is 60.94%. When $K = 2$, IVA-M-EMK outperforms IVA-SPICE in terms of the F1-score, except when the text and Image2text modalities are fused. As K increases, IVA-SPICE effectively exploits sparsity through the inverse covariance matrix (precision matrix) and, thus, reduces the effects of confounding joint features yielding better classification performance in all cases. In addition, while the superiority of IVA-M-EMK is evident when working with two modalities, as we can see from Table 2 it does not persist as the number of modalities analyzed increases. As the dimensionality of the probability space expands with an increase in the number of modalities (K), the estimation of the $K-$dimensional probability densities that correspond to each estimated SCV becomes a challenging task. This is where IVA-SPICE becomes crucial in the joint feature generation process. IVA-SPICE mitigates the effects of confounding joint features, resulting in improved classification performance compared to IVA-M-EMK. Hence, IVA-SPICE emerges as a valuable approach for handling the complexities introduced by higher-dimensional probability spaces and multiple modalities.

Table 2 Performance comparison in terms of F1-score for different number of modalities fused to train the classification model

Modalities (K)	IVA	F1
$K = 2$: Text and Image	**IVA-M-EMK**	**77.45%**
	IVA-SPICE	67.92%
$K = 2$: Text and Topic Modeling	**IVA-M-EMK**	**57.25%**
	IVA-SPICE	53.45%
$K = 2$: Text and Image2text	IVA-M-EMK	51.40%
	IVA-SPICE	**56.97%**
$K = 2$: Image and Topic Modeling	**IVA-M-EMK**	**72.59%**
	IVA-SPICE	63.25%
$K = 2$: Image and Image2text	**IVA-M-EMK**	**73.45%**
	IVA-SPICE	66.37%
$K = 2$: Topic Modeling and Image2text	**IVA-M-EMK**	**53.20%**
	IVA-SPICE	48.71%
$K = 3$: Text, Image, and Image2text	IVA-M-EMK	61.73%
	IVA-SPICE	**69.43%**
$K = 3$: Text, Image, and Topic Modeling	IVA-M-EMK	58.43%
	IVA-SPICE	**67.99%**
$K = 3$: Text, Image2text, and Topic Modeling	IVA-M-EMK	55.71%
	IVA-SPICE	**62.95%**
$K = 3$: Image, Image2text, and Topic Modeling	IVA-M-EMK	62.07%
	IVA-SPICE	**68.58%**
$K = 4$: Text, Image, Image2text, and Topic Modeling	IVA-M-EMK	61.22%
	IVA-SPICE	**73.01%**

When $K = 2$, we combine $\frac{4!}{2!2!}$ modalities, similar for $K = 3$

Bold values denote the highest level of performance

5 Conclusion

This study highlights five interesting directions that can be explored in future work. First, although the explainability of IVA has been addressed in Damasceno et al. (2022), in future work, we plan to create formal settings where humans can evaluate whether a set of extracted features has human-identifiable semantic coherence. These quantitative methods have been similarly used for measuring semantic meaning in inferred topics (Chang et al., 2009). By developing human-based evaluation metrics, we will not only be able to assess the IVA joint representation space, but more importantly, we will be able to identify potential biases related to specific characteristics of the collected social media posts, enabling us to correct our model before it is deployed at scale. Second, IVA-M-EMK and IVA-SPICE can be integrated into more sophisticated and complex classification systems such as those based on deep neural networks to further improve classification performance. The flexibility to integrate IVA algorithms into any classification system stems from the fact that the proposed IVA algorithms are unsupervised methods and are independent of the choice of the classification algorithm. Third, as mentioned previously, future work includes further optimization of the method to provide linear scalability on cluster computing

environments, exploiting distributed CPU and GPU-based programming frameworks, as well as implementing parallel versions of all IVA algorithms. Fourth, we are interested in incorporating additional modalities into our study since, as we demonstrated in this work, the multivariate data fusion model based on IVA-SPICE provides enhanced detection performance as the number of modalities increases. Examples of additional modalities include those based on videos or metadata. Last, we propose to study the convergence of our new IVA algorithms and derive the optimal conditions for both IVA-M-EMK and IVA-SPICE on the space of probability density functions.

Author contributions LPD: Data Curation, Visualization, Investigation, Methodology, Software, Writing—ER: Data Curation, Investigation, Methodology, Software—AS: Data Curation, Investigation, Software (Data processing and feature engineering), Writing (Review & Editing)—IW: Software (Parallel implementation), NJ: Resources, Validation, Writing (Review & Editing)—CCC, RC, and ZB: Conceptualization, Supervision, Validation, Methodology, Project Administration, Writing (Review & Editing).

Funding The authors acknowledge the support of Capes Finance Code 001, CNPq Proc. 313151/2020-2, and Funcap Grant PS-0186‒00103.01.00/21.

Data availability The data and materials to reproduce the experiments are available at the following repository (https://zoisboukouvalas.github.io/Code.html).

Code availability The code of the proposed method is available at the following repository (https://zoisboukouvalas.github.io/Code.html).

Declarations

Conflict of interest All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest in the subject matter or materials discussed in this manuscript.

Consent for publication This study does not involve human subjects or any sensitive data.

Ethical approval Not applicable.

Consent to participate This study does not involve human subjects or any sensitive data.

References

Adalı, T., Anderson, M., & Fu, G. S. (2014). Diversity in independent component and vector analyses: Identifiability, algorithms, and applications in medical imaging. *IEEE Signal Processing Magazine, 31*(3), 18–33.

Amari, S. i., & Douglas, S. C. (1998). Why natural gradient?. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing* (Vol. 2, pp. 1213–1216). IEEE.

Anderson, M., Adalı, T., & Li, X. L. (2012). Joint blind source separation with multivariate gaussian model: Algorithms and performance analysis. *IEEE Transactions on Signal Processing, 60*(4), 1672–1683. https://doi.org/10.1109/TSP.2011.2181836

Balakrishnan, S., VanGessel, F. G., Boukouvalas, Z., Barnes, B. C., Fuge, M. D., & Chung, P. W. (2021). Locally optimizable joint embedding framework to design nitrogen-rich molecules that are similar but improved. *Molecular Informatics, 40*, 2100011.

Baltrušaitis, T., Ahuja, C., & Morency, L. P. (2018). Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 41*(2), 423–443.

Banerjee, O., Ghaoui, L., & dAspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *Journal of Machine Learning Research, 9*, 485–516.

Boididou, C., Papadopoulos, S., Zampoglou, M., Apostolidis, L., Papadopoulou, O., & Kompatsiaris, I. (2018). Detection and visualization of misleading content on Twitter. *International Journal of Multimedia Information Retrieval*. https://doi.org/10.1007/s13735-017-0143-x

Boididou, C., Papadopoulos, S., Zampoglou, M., Apostolidis, L., Papadopoulou, O., & Kompatsiaris, Y. (2018). Detection and visualization of misleading content on Twitter. *International Journal of Multimedia Information Retrieval, 7*(1), 71–86. https://doi.org/10.1007/s13735-017-0143-x

Boukouvalas, Z. (2018). Development of ICA and IVA algorithms with application to medical image analysis. *arXiv preprint*arXiv:1801.08600.

Boukouvalas, Z., Fu, G. S., & Adalı, T. (2015). An efficient multivariate generalized Gaussian distribution estimator: Application to IVA. In *2015 49th Annual Conference on Information Sciences and Systems (CISS)* (pp. 1–4). IEEE.

Boukouvalas, Z., Levin-Schwartz, Y., Mowakeaa, R., Fu, G. S., & Adalı, T. (2018). Independent component analysis using semi-parametric density estimation via entropy maximization. In *2018 IEEE Statistical Signal Processing Workshop (SSP)* (pp. 403–407). IEEE.

Boukouvalas, Z., Mallinson, C., Crothers, E., Japkowicz, N., Piplai, A., Mittal, S., Joshi, A., & Adalı, T. (2020) Independent component analysis for trustworthy cyberspace during high impact events: An application to covid-19. arxiv:2006.01284

Boukouvalas, Z., Puerto, M., Elton, D. C., Chung, P. W., & Fuge, M. D. (2021). Independent vector analysis for molecular data fusion: Application to property prediction and knowledge discovery of energetic materials. In *2020 28th European Signal Processing Conference (EUSIPCO)* (pp. 1030–1034). IEEE.

Cao, J., Qi, P., Sheng, Q., Yang, T., Guo, J., & Li, J. (2020). Exploring the role of visual content in fake news detection. In *Disinformation, misinformation, and fake news in social media* (pp. 141–161)

Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., & Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In *Advances in Neural Information Processing Systems* (pp. 288–296)

Cichocki, A., & Yang, H. (1996). A new learning algorithm for blind signal separation. *Advances in Neural Information Processing Systems, 8*, 757–763.

Comon, P., & Jutten, C. (2010). *Handbook of blind source separation: Independent component analysis and applications*. Academic Press.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273–297. https://doi.org/10.1023/A:1022627411411

Damasceno, L. P., Cavalcante, C. C., Adalı, T., & Boukouvalas, Z. (2021). Independent vector analysis using semi-parametric density estimation via multivariate entropy maximization. In *ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 3715–3719). IEEE.

Damasceno, L. P., Shafer, A., Japkowicz, N., Cavalcante, C. C., & Boukouvalas, Z. (2022). Efficient multivariate data fusion for misinformation detection during high impact events. In *Discovery Science: 25th International Conference, DS 2022, Montpellier, France, 10–12 Oct, 2022, Proceedings* (pp. 253–268). Springer.

Dempster, A. (1972). Covariance selection. *Biometrics, 28*(1), 157–175.

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. *CoRR*arxiv:1810.04805

Dick, J., Kuo, F. Y., & Sloan, I. H. (2013). High-dimensional integration: The quasi-Monte Carlo way. *Acta Numerica, 22*, 133–288. https://doi.org/10.1017/S0962492913000044

Friedman, J., Hastie, T., & Tibshirani, R. (2007). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics, 9*(3), 432–441.

Fu, G., Boukouvalas, Z., & Adalı, T. (2015). Density estimation by entropy maximization with kernels. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 1896–1900). https://doi.org/10.1109/ICASSP.2015.7178300

Hyvärinen, A., Karhunen, J., & Oja, E. (2004). *Independent component analysis* (Vol. 46). John Wiley & Sons.

Kim, T., Eltoft, T., & Lee, T. W. (2006). Independent vector analysis: An extension of ICA to multivariate components. *Independent component analysis and blind signal separation* (pp. 165–172). Springer.

Lauritzen, S. (1996). *Graphical models*. Clarendon Press.

Li, X. L., & Adalı, T. (2010). Independent component analysis by entropy bound minimization. *IEEE Transactions on Signal Processing, 58*(10), 5151–5164.

Mikolov, T., Chen, K., Corrado, G. S., & Dean, J. (2013). Efficient estimation of word representations in vector space. *CoRR*arXiv:1301.3781

Moroney, C., Crothers, E., Mittal, S., Joshi, A., Adalı, T., Mallinson, C., Japkowicz, N., & Boukouvalas, Z. (2021). The case for latent variable vs deep learning methods in misinformation detection: An application to covid-19. In: *International Conference on Discovery Science* (pp. 422–432). Springer.

Niederreiter, H. (1992). *Random number generation and quasi-Monte Carlo methods*. Society for Industrial and Applied Mathematics.

Ramachandram, D., & Taylor, G. W. (2017). Deep multimodal learning: A survey on recent advances and trends. *IEEE Signal Processing Magazine, 34*(6), 96–108.

Sharma, K., Qian, F., Jiang, H., Ruchansky, N., Zhang, M., & Liu, Y. (2019). Combating fake news: A survey on identification and mitigation techniques. *ACM Transactions on Intelligent Systems and Technology (TIST), 10*(3), 1–42.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, 58*(1), 267–288.