Semi-Supervised Multi-aspect Detection of Misinformation using Hierarchical Joint Decomposition

Sara Abdali[1],
Neil Shah[2], and
Evangelos E. Papalexakis[1]

[1] Department of Computer Science and Engineering University of California
Riverside 900 University Avenue, Riverside, CA, USA
sabda005@ucr.edu epapalex@cs.ucr.edu
[2] Snap Inc. nshah@snap.com

Abstract. Distinguishing between misinformation and real information is one of the most challenging problems in today's interconnected world. The vast majority of the state-of-the-art in detecting misinformation is fully supervised, requiring a large number of high-quality human annotations. However, the availability of such annotations cannot be taken for granted, since it is very costly, time-consuming, and challenging to do so in a way that keeps up with the proliferation of misinformation. In this work, we are interested in exploring scenarios where the number of annotations is limited. In such scenarios, we investigate how to tap into a diverse number of resources that characterize a news article, henceforth referred to as "aspects" can compensate for the lack of labels. In particular, our contributions in this paper are twofold: 1) We propose the use of three different aspects: article content, context of social sharing behaviors, and host website/domain features, and 2) We introduce a principled tensor based embedding framework that combines all those aspects effectively. We propose `HiJoD` a 2-level decomposition pipeline which not only outperforms state-of-the-art methods with F1-scores of 74% and 81% on `Twitter` and `Politifact` datasets respectively, but also is an order of magnitude faster than similar ensemble approaches.

Keywords: Misinformation Detection · Hierarchical Tensor Decomposition · Multi-aspect Modeling · Ensemble Learning · Semi-Supervised Classification

1 Introduction

In recent years, we have experienced the proliferation of websites and outlets that publish and perpetuate misinformation. With the aid of social media platforms, such misinformation propagates wildly and reaches a large number of the population, and can, in fact, have real-world consequences. Thus, understanding and flagging misinformation on the web is an extremely important and timely problem, which is here to stay.

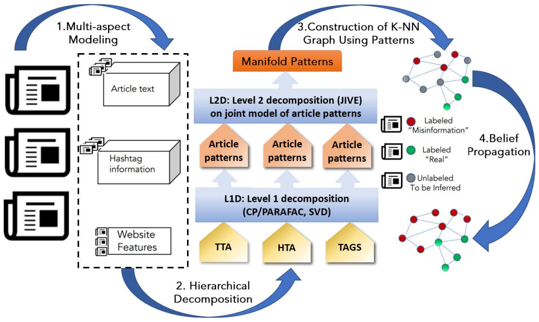

Fig. 1: Overview: We propose using three aspects1: a) content, b) social context (in the form of hashtags), and c) features of the website serving the content. We, further, propose `HiJoD` hierarchical approach for finding latent patterns derived from those aspects, generate a graphical representation of all articles in the embedding space, and conduct semi-supervised label inference of unknown articles.

There have been significant advances in detecting misinformation from the article content, which can be largely divided into knowledge-based fact-checking and style-based approaches; [18,28] survey the landscape. Regardless of the particular approach followed, the vast majority of the state-of-the-art is fully supervised, requiring a large number of high-quality human annotations in order to learn the association between content and whether an article is misinformative. For instance, in [3], a decision-tree based algorithm is used to assess the credibility of a tweet, based on Twitter features. In another work, Rubin et al. [26] leverage linguistic features and a SVM-based classifier to find misleading information. Similarly, Horne and Adali [10] apply SVM classification on content-based features. There are several other works [6,13,12] for assessing credibility of news articles, all of which employ propagation models in a supervised manner. Collecting human annotation for misinformation detection is a complicated and time consuming task, since it is challenging and costly to identify human experts who can label news articles devoid of their own subjective views and biases, and possess all required pieces of information to be a suitable "oracle." However, there exist crowd-sourced schemes such as the browser extension "BS Detector" [3] which provide coarse labels by allowing users to flag certain articles as different types of misinformation, and subsequently flagging the entire source/domain as the majority label. Thus, we are interested in investigating methods that can compensate for the lack of large amounts of labels with leveraging different signals or aspects that pertain to an article. A motivating consideration is that we as humans empirically consider different aspects of a particular article in order to distinguish between misinformation and real information. For example, when

[3] http://bsdetector.tech/

we review a news article on a web site, and we have no prior knowledge about the legitimacy of its information, we not only take a close look at the content of the article but also we may consider how the web page looks (e.g. does it look "professional"? Does it have many ads and pop-up windows that make it look untrustworthy?). We might conclude that untrustworthy resources tend to have more "messy" web sites in that they are full of ads, irrelevant images, pop-up windows, and scripts. Most prior work in the misinformation detection space considers only one or two aspects of information, namely content, headline or linguistic features. In this paper, we aim to fill that gap by proposing a comprehensive method that emulates this multi-aspect human approach for finding latent patterns corresponding to different classes, while at the same time leverages scarce supervision in order to turn those insights into actionable classifiers. In particular, our proposed method combines multiple aspects of article, including (a) article content (b) social sharing context and (c) source webpage context each of which is modeled as a tensor/matrix. The rationale behind using tensor based model rather than state-of-the-art approaches like deep learning methods is that, such approaches are mostly supervised methods and require considerable amount of labeled data for training. On the contrary, we can leverage tensor based approaches to find meaningful patterns using less labels. Later on, we will compare the performance of tensor-based modeling against deep learning methods when there is scarcity of labels. We summarize the contributions as follows:

- **Leveraging different aspects of misinformation**: In this work, we not only consider content-based information but also we propose to leverage multiple aspects for discriminating misinformative articles. In fact, we propose to create multiple models, each of which describes a distinct aspect of articles.
- **A novel hierarchical tensor based ensemble model**: We leverage a hierarchical tensor based ensemble method i.e., HiJoD to find manifold patterns that comprise multi-aspect information of the data.
- **Evaluation on real data**: We extensively evaluate HiJoD on two real world datasets i.e., Twitter and Politifact. HiJoD not only outperforms state-of-the-art alternatives with F1 score of 74% and 81% on above datasets respectively, but also is significantly faster than similar ensemble methods. We make our implementation publicly available[4] to promote the reproducibility.

2 Problem Formulation

Considering the following formulation of misinformation detection problem:

> **Given** N articles with associated 1) article text, social sharing context (hashtags that are used when sharing the article) 2) HTML source of the article's publisher webpage, and, 3) binary (misinformative/real) labels for p% of articles, **Classify** the remaining articles into the two classes.

[4] https://github.com/Saraabdali/HiJoD-ECMLPKDD

At a high level, we aim to demonstrate the predictive power of incorporating multiple aspects of article content and context on the misinformation detection task, and consider doing so in a low-label setting due to practical challenges in data labeling for this task. Our proposed method especially focuses on (a) multi-aspect data modeling and representation choices for downstream tasks, (b) appropriate triage across multiple aspects, and (c) utility of a semi-supervised approach which is ideal in sparse label settings. In the next section, we detail our intuition and choices for each of these components.

We aim to develop a manifold approach which can discriminate misinformative and real news articles by leveraging content-based information in addition to other article aspects i.e. social context and publisher webpage information. To this end, we propose using tensors and matrices to analyze these aspects jointly. We develop a three-stage approach: (a) *multi-aspect article representation*: we first introduce three feature context representations (models) which describe articles from different points of view (aspects), (b) *Manifold patterns finding using a hierarchical approach*: In proposed `HiJoD`, we first decompose each model separately to find the latent patterns of the articles with respect to the corresponding aspect, then we use a strategy to find shared components of the individual patterns (c) *semi-supervised article inference*: finally, we focus on the inference task by construing a K-NN graph over the resulted manifold patterns and propagating a limited set of labels.

3 Proposed Methodology

3.1 Multi-Aspect Article Representation

We first model articles with respect to different aspects. We suggest following tensors/matrix to model content, social context and source aspects respectively:

- **(Term×Term×Article) Tensor** (`TTA`): The most straightforward way that comes into mind for differentiating between a fake news and a real one is to analyze the content of the articles. Different classes of news articles, i.e., fake and real classes tend to have some common words that co-occur within the text. Thus, we use a tensor proposed by [11] to model co-occurrence of these common words. This model not only represents content-based information but also considers the relations between the words and is stronger than widely used bag of words and tf/idf models. To this end, we first create a dictionary of all articles words and then slide a window across the text of each article and capture the co-occurred words. As a result, we will have a co-occurrence matrix for each article. By stacking all these matrices, we create a three mode tensor where the first two dimensions correspond to the indices of the words in the dictionary and the third mode indicates the article's ID. We may assign binary values or frequency of co-occurrence to entries of the tensors. However, as shown in [2] binary tensor is able to capture more nuance patterns. So, in this work we also use binary values.
- **(Hashtag×Term×Article) Tensor** (`HTA`): Hashtags often show some trending across social media. Since, hashtags assigned to an article usually, convey

social context information which is related to content of news article, we propose to construct a hashtag-content tensor to model such patterns. In this tensor, we want to capture co-occurrence of words within the articles and the hashtags assigned to them. For example, an article tagged with a hashtag `#USElection2016` probably consists of terms like “Donald Trump” or “Hillary Clinton”. These kind of co-occurrences are meaningful and convey some patterns which may be shared between different categories of articles. The first two modes of this tensor correspond to hashtag and word indices respectively and the third mode is article mode.

- **(Article×HTML features) Matrix (`TAGS`)**: Another source of information is the trustworthiness of serving webpage. In contrast to reliable web resources which usually have a standard form, misleading web pages may often be messy and full of different advertisement, pop-ups and multimedia features such as images and videos. Therefore, we suggest to create another model to capture the look and the feel of the web page serving the content of news article. We can approximate look of a web page by counting HTML features and tags and then represent it by a (article, HTML feature) matrix. The rows and columns of this matrix indicate the article and hashtag IDs respectively. We fill out the entries by frequency of HTML tags in the web source of each article domain. Fig. 2 demonstrates aforementioned aspects.

3.2 Hierarchical Decomposition

Now, the goal is to find manifold patterns with respect to all introduced aspects. To this end, we look for shared components of the latent patterns.

Level-1 decomposition: Finding article patterns with respect to each aspect As mentioned above, first, we find the latent patterns of the articles with respect to each aspect. To do so, we decompose first two tensor models using Canonical Polyadic or CP/PARAFAC decomposition [8] into summation of rank one tensors, representing latent patterns within the tensors as follows:

$$\mathcal{X} \approx \Sigma_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \tag{1}$$

where $\mathbf{a}_r \in \mathbb{R}^I$, $\mathbf{b}_r \in \mathbb{R}^J$, $\mathbf{c}_r \in \mathbb{R}^K$ [24] and R is the rank of decomposition. The factor matrices are defined as $\mathbf{A} = [\mathbf{a}_1 \ \mathbf{a}_2 \ldots \mathbf{a}_R]$, $\mathbf{B} = [\mathbf{b}_1 \ \mathbf{b}_2 \ldots \mathbf{b}_R]$, and $\mathbf{C} = [\mathbf{c}_1 \ \mathbf{c}_2 \ldots \mathbf{c}_R]$ where for `TTA` and `HTA`, $\mathbf{C}$ corresponds to the article mode and comprises the latent patterns of the articles with respect to content and social context respectively. The optimization problem for finding factor matrices is:

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C}} = \left\| \mathcal{X} - \Sigma_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \right\|^2 \tag{2}$$

To solve the above optimization problem, we use Alternating Least Squares (ALS) approach which solves for any of the factor matrices by fixing the others [24]. For the tag matrix, to keep the consistency of the model we suggest to use SVD decomposition ($\mathbf{X} \approx \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$) because CP/PARAFAC is one extension of SVD for higher mode arrays. In this case, $\mathbf{U}$ comprises the latent patterns of articles with respect to overall look of the serving webpage.

Level-2 decomposition: Finding manifold patterns Now, we want to put together article mode factor matrices resulted from individual decompositions to find manifold patterns with respect to all aspects. Let's $\mathbf{C_{TTA}} \in \mathbb{R}^{N \times r_1}$ and $\mathbf{C_{HTA}} \in \mathbb{R}^{N \times r_2}$ be the third factor matrices resulted from CP decomposition of TTA and HTA respectively and let's $\mathbf{C_{TAGS}} \in \mathbb{R}^{N \times r_3}$ be $\mathbf{U}$ matrix resulted from rank r_3 SVD of TAGS where N is the number of articles. Since all these three matrices comprise latent patterns of the articles, we aim at finding patterns which are shared between all. To do so, we concatenate the three article embedding matrices ($\mathbf{C}$) and decompose the joint matrix of size $(r_1 + r_2 + r_3) \times N$ to find shared components. Like level-1 decomposition, we can simply take the SVD of joint matrix. SVD seeks an accurate representation in least-square setting but to find a meaningful representation we need to consider additional information i.e., the relations between the components. Independent components analysis (ICA) tries to find projections that are statistically independent as follows:

$$[\mathbf{C_{TTA}}; \mathbf{C_{HTA}}; \mathbf{C_{TAGS}}] = \mathbf{AS} \tag{3}$$

Where $\mathbf{A}$ is the corresponding shared factor of size $(r_1+r_2+r_3) \times N$ and $\mathbf{S}$ is the mixed signal of size $(r_1 + r_2 + r_3) \times R$. To achieve the statistical independence, it looks for projections that leads to projected data to be as far from Gaussian distribution as possible. The problem is that, SVD (PCA) and ICA may result in identical subspaces. In fact, this happens when the direction of greatest variation and the independent components span the same subspace [29]

In order to consider relations between components and find a meaningful representation, we can consider shared and unshared components for $\mathbf{C}$ matrices such that unshared components are orthogonal to shared ones. To this end, we propose to use the Joint and Individual Variation Explained (JIVE) for level-2 decomposition[22]. More precisely, let's consider $\mathbf{A_i}$ and $\mathbf{J_i}$ to be the matrices representing the individual structure and submatrix of the joint pattern for $i \in \{\mathtt{HTA}, \mathtt{TTA}, \mathtt{TAGS}\}$ respectively such that they satisfy the orthogonality constraint. Using JIVE method, we decompose each article mode factor matrix $\mathbf{C}$ as follows:

$$\mathbf{C_i} = \mathbf{J_i} + \mathbf{A_i} + \epsilon_i \tag{4}$$

To find $\mathbf{A}$ and $\mathbf{J}$ matrices, we use the approach presented in [22]. In other words, we fix $\mathbf{A}$ and find $\mathbf{J}$ that minimizes the following residual matrix:

$$\|\mathbf{R}\|^2 = \|\epsilon_{TTA}; \epsilon_{HTA}; \epsilon_{Tags}\|^2 \tag{5}$$

The joint structure $\mathbf{J}$ which minimizes $\|\mathbf{R}\|^2$ is equal to the rank r SVD of joint matrix when we remove the individual structure and in the same way individual structure for each $\mathbf{C}$ matrix is the rank r_i SVD of $\mathbf{C}$ matrix when we remove the joint structure [22]. Fig. 2 and Algorithm. 1 demonstrate the details.

3.3 Semi-Supervised Article Inference

Previous step, provides us with a $N \times r$ matrix which comprises manifold patterns of N articles. In this step, we leverage this matrix to address the semi-supervised

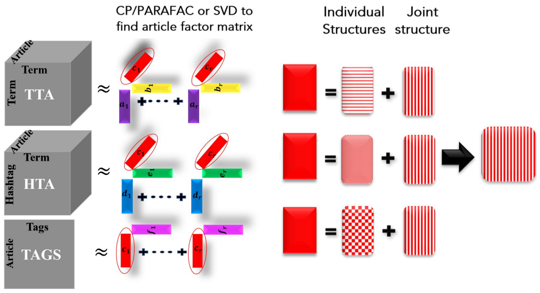

Fig. 2: `HiJoD` finds manifold patterns of the articles that can be used for classification.

Algorithm 1: `HiJoD` Hierarchical decomposition

```
Input: TTA, HTA, TAGS
Output: $\mathbf{J_{joint}}$
First level of Decomposition
$\mathbf{C_{TTA}}$ =CP-ALS (TTA,$r_1$);
$\mathbf{C_{HTA}}$ =CP-ALS (HTA,$r_2$);
$\mathbf{C_{Tags}}$ =SVD (TAGS,$r_3$);
$\mathbf{C^{Joint}} = [\mathbf{C_{TTA}}; \mathbf{C_{HTA}}; \mathbf{C_{Tags}}]$
Second level of Decomposition
while $\|\mathbf{R}\|^2 < \epsilon$ do
    $\mathbf{J} = \mathbf{U\Sigma V}^T$
    $[\mathbf{J_{TTA}}; \mathbf{J_{HTA}}; \mathbf{J_{Tags}}]$ =SVD $(\mathbf{C_{Joint}}, r_{joint})$//Calculate $r_{joint}$ using Algorithm 2
    for $i \in \{$ HTA,TTA,TAGS $\}$ do
        $\mathbf{A_i} = \mathbf{C_i} - \mathbf{J_i}$
        $\mathbf{A_i}$ =SVD $(\mathbf{A_i} * (\mathbf{I} - \mathbf{VV}^T), r_i)$ //To satisfy the orthogonality constraint
    end
    $\mathbf{C_i}^{new} = \mathbf{C_i}^{joint} - \mathbf{A_i}$
    $\epsilon_i = |\mathbf{C_i}^{joint} - \mathbf{C_i}^{new}|$
    $\mathbf{C}^{joint} = \mathbf{C}^{new}$
end
```

Algorithm 2: Calculating the rank for joint and individual matrices

```
Input: $\alpha \in (0, 1)$ , n_perm
Output: r
Let's $\lambda_j$ be the $j$'th singular value of $\mathbf{X}, i = 1, \ldots, rank(\mathbf{X})$.
while $n \leq n\_perm$ do
    Permute the columns within each $X_i$, and calculate the singular values of the resulting
      $\mathbf{C_{joint}}$
end
$\lambda_i^{perm} = 100(1 - \alpha)$ percentile of $j$'th singular values
Choose largest $r$ such that $\forall j \leq r, \lambda_j \geq \lambda_j^{perm}$
```

problem of classifying misinformation. Row i of this matrix represents article i in r dimensional space, we suggest to construct a K-NN graph using this matrix such that each node represents an article and edges are Euclidean distances between articles (rows of manifold patterns matrix) to model the similarity of articles. We utilize the Fast Belief Propagation (FaBP) algorithm as is described in [2,17] to propagate labels of known articles (fake or real) throughout K-NN graph. Belief propagation is a message passing-based algorithm. Let's $m_{j\hookrightarrow i}(x_i)$ indicates the message passes from node j to node i and N_i denotes all the neighboring nodes of node i. $m_{j\hookrightarrow i}(x_i)$ conveys the opinion of node j about the belief (label) of node i. Each node of a given graph G leverages messages received from neighboring nodes and calculates its belief iteratively as follows:.

$$b_i(x_i) \propto \prod_{j\in(N_i)} m_{j\hookrightarrow i}(x_i) \tag{6}$$

4 Experimental Evaluation

In this section, we discuss the datasets, baselines and experimental results.

4.1 Dataset Description

`Twitter` **dataset** To evaluate `HiJoD`, we created a new dataset by crawling Twitter posts contained links to articles and shared between June and August 2017. This dataset comprises 174k articles from more than 652 domains. For labeling the articles, we used BS-Detector, which is a *crowd-sourced* toolbox in form of a browser extension, as ground truth. BS-Detector categorizes domains into different categories such as bias, clickbait, conspiracy, fake, hate, junk science, rumor, satire, and unreliable. We consider above categories as "misinformative" class. A key caveat behind BS-Detector, albeit being the most scalable and publicly accessible means of labeling articles at-large, is that labels actually pertains to the *domain* rather than the article itself. At the face of it, this sounds like the labels obtained are for an entirely different task, however, Helmstetter et al. [9] show that training for this "weakly labeled" task (using labels for the domains), and subsequently testing on labels pertaining to the articles, yields minimal loss in accuracy and labels still hold valuable information. Thus, we choose BS-Detector for ground truth. However, in order to make our experiments as fair as possible, in light of the above fact regarding the ground truth, we do as follows:

- We restrict the number of articles per domain we sample into our pool of articles. So, we experimented with randomly selecting a single article per domain and iterating over 100 such sets. Since we have 652 different domains in `Twitter` dataset, in each iteration we chose 652 non-overlapping articles so, totally we examine different approaches for 65.2K different articles.
- In order to observe the effect of using more articles per domain, we repeated the experiments for different number of articles per domain. We observed that the embedding that uses HTML tags receives a disproportionate boost in its performance. We attribute this phenomenon to the fact that all instances coming from the same domain have exactly the same HTML features, thus classifying correctly one of them implies correct classification for the rest, which is proportional to the number of articles per domain.

- We balance the dataset so that we have 50% fake and 50% real articles at any given run per method. We do so to have a fair evaluation setting and prevent the situation in which there is a class bias. To show the insensitivity of HiJoD to class imbalance, we also experiment on an imbalanced dataset.

Politifact **dataset**: For second dataset, we leverage FakeNewsNet dataset that the authors of [14] used for their experiments[5][28,15]. This dataset consists of 1056 news articles from the Politifact fact checking website, 60% being real and 40% being fake. Using this imbalanced dataset, we can experiment how working on an imbalanced dataset may affect the proposed approach. Since not all of the signals we used from Twitter dataset exist in Politifact dataset, For this dataset we created the following embeddings:

- TTA tensor: To keep the consistency, we created TTA from articles text.
- User-News Interaction Embedding: We create a matrix which represent the users who tweets a specific news article, as proposed in [14].
- Publisher-News Interaction Embedding: We create another matrix to show which publisher published a specific news article, as proposed in [14].

Using the aforementioned signals, we can also test the efficacy of HiJoD when we leverage aspects other than those we proposed in this work.

4.2 Baselines for Comparison

As mentioned earlier, the two major contributions of HiJoD are: 1) the introduction of different aspects of an article and how they influence our ability to identify misinformation more accurately, and 2) how we leverage different aspects to find manifold patterns which belongs to different classes of articles. Thus, we conduct experiments with two categories of baseline to test each contribution separately:

Content-based approaches to test the effect of additional aspects. We compare with state-of-the-art content-based approaches to measure the effect of introducing additional aspects (hashtags and HTML features) into the mix, and whether the classification performance improves. We compare against:

- TTA/BP In [2], Bastidas et al. effectively use the co-occurrence tensor in a semi-supervised setting. They demonstrate how this tensor embedding outperforms other purely content-based state-of-the-art methods such as SVM on content-based features and Logistic regression on linguistic features [7,10]. Therefore, we select this method as the first baseline, henceforth referenced as "TTA". The differences between our results and the results reported by Bastidas et al [2] is due to using different datasets. However, since we used the publicly available code by Bastidas et al. [6] [2], if we were to use the same data in [2] the results would be exactly the same.
- tf_idf/SVM is the well-known term frequency–inverse document frequency method widely used in text mining and information retrieval and illustrates

[5] https://github.com/KaiDMML/FakeNewsNet

[6] https://github.com/Saraabdali/Fake-News-Detection-˙ASONAM-2018

how important a word is to a document. We create a `tf_idf` model out of articles text and apply SVM classifier on the resulted model.

- `Doc2Vec/SVM` is an NLP toolbox proposed by Le et al.[20] from Google. This model is a shallow, two-layer neural network that is trained to reconstruct linguistic contexts of document. This algorithm is an extension to word2vec which can generate vectors for words. Since the SVM classifier is commonly used on this model, we also leverage SVM for document classification.[7]
- `fastText` is an NLP library by Facebook Research that can be used to learn word representations to efficiently classify document. It has been shown that `fastText` results are on par with deep learning models in terms of accuracy but an order of magnitude faster in terms of performance.[8]
- `GloVe/LSTM` GloVe is an algorithm for obtaining vector representations of the words. Using an aggregated global word-word co-occurrence, this method results in a linear substructures of the word vector space. We use the method proposed in [21,19] and we create a dictionary of unique words and leverage Glove to map indices of words into a pre-trained word embedding. Finally, as suggested, we use LSTM to classify articles.[9]

Ensemble approaches to test the efficacy of our fused method. Another way to jointly derive the article patterns, specially when the embeddings are tensors and matrices, is to couple matrices and tensors on shared mode(s) i.e., article mode in this work. In this technique which called coupled Matrix and Tensor Factorization (`CMTF`), the goal is to find optimized factor matrices by considering all different optimization problems we have for individual embeddings. Using our proposed embeddings, we the optimization problem is:

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C},\mathbf{D},\mathbf{E},\mathbf{F}} \|\mathtt{TTA} - [\mathbf{A},\mathbf{B},\mathbf{C}]\|^2 + \|\mathtt{HTA} - [\mathbf{D},\mathbf{E},\mathbf{C}]\|^{\mathbf{2}} + \|\mathtt{TAGS} - \mathbf{C}\mathbf{F}^{\mathbf{T}}\|^{\mathbf{2}} \quad (7)$$

where $[\mathbf{A},\mathbf{B},\mathbf{C}]$ denotes $\Sigma_{r=1}^{R}\mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$, and $\mathbf{C}$ is the shared article mode as shown in Fig.3. In order to solve the optimization problem above, we use the approach introduced in [5], which proposes *all-at-once-optimization* by computing the gradient of every variable of the problem, stacking the gradients into a long vector, and applying gradient descent. There is an advanced version of coupling called `ACMTF` that uses weights for rank-one components to consider both shared and unshared ones[1]. We applied `ACMTF` as well and it led to similar results, however, slower than standard `CMTF`. Thus, we just report the result of `CMTF`.

Moreover, as mentioned earlier, to derive the manifold pattern which is shared between the aspects, we can leverage different mathematical approaches such as:

- `SVD` (Singular Value Decomposition)
- `JICA` (Joint Independent Component Analysis)

on $\mathbf{C_{joint}}$. To measure the performance of `JIVE` method for finding manifold patterns, we will also examine the above approaches for level-2 decomposition.

[7] https://github.com/seyedsaeidmasoumzadeh/Binary-Text-Classification-Doc2vec-SVM

[8] https://github.com/facebookresearch/fastText

[9] https://github.com/prakashpandey9/Text-Classification-Pytorch

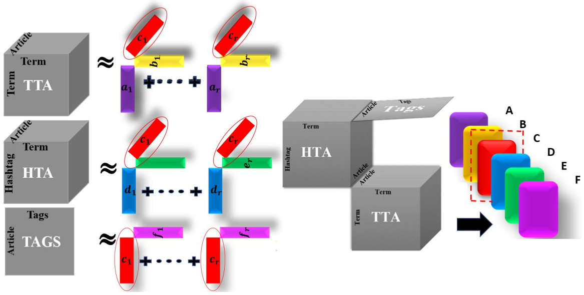

Fig. 3: Couple Tensor Matrix Factorization for finding shared patterns of the articles.

4.3 Comparing with Baselines

Implementation. We implemented `HiJoD` and all other approaches in MATLAB using Tensor Toolbox version 2.6. Moreover, to implement `JICA` approach, we used FastICA[10] , a fast implementation of ICA. For the baseline approach `CMTF` we used the Toolbox[11] in [5,4]. For the belief propagation step, we used implementation introduced in [17]. Based on the experiments reported in [2], we employed a sliding window of size 5 for capturing the co-occurring words. Using AutoTen [25], which finds the best rank for tensors, we found out the best rank for `TTA` and `HTA` is 10 and 40 respectively. For the `TAGS` embedding, we took the full SVD to capture the significant singular values and set the `TAGS` SVD rank to 20. Since for `CMTF` approach we have to choose the same rank, as a heuristic, we used the range of ranks for individual embeddings (10 for `HTA` and 40 for `TTA`) and again searched for the ensemble rank in this range. Based on our experiments, rank 30 leads to the best results in terms of F1-score. Moreover, grid search over 1-30 nearest neighbors yielded choice of 5 neighbors for `CMTF` and 15 for `HiJoD`. For the rank of joint model as well as individual and joint structures i.e., $\mathbf{A_i}$ and $\mathbf{J_i}$, we used the strategy proposed in [22] and for reproducibility purposes is demonstrated in Algorithm 2. The intuition is to find the rank of joint structure i by comparing the singular values of the original matrix with the singular values of n_{perm} randomly permuted matrices. If the j^{th} singular value in the original matrix is $\geq 100(1-\alpha)$ percentile of the j^{th} singular value of n_{perm} matrices, we keep it as a significant one. Number of these significant singular values shows the rank. n_{perm} and α are usually set to 100 and 0.05 respectively. For timing experiment, we used the following configuration:

- CPU: Intel(R) Core(TM) i5-8600K CPU @3.60GHz
- OS: CentOS Linux 7 (Core)
- RAM: 40GB

[10] https://github.com/aludnam/MATLAB/tree/master/FastICA_25

[11] http://www.models.life.ku.dk/joda/CMTF_Toolbox

%Labels	tf__idf/SVM	Doc2Vec/SVM	fastText	GloVe/LSTM	TTA/BP	HiJoD
10	0.500±0.032	0.571±0.092	0.562±0.031	-	0.582±0.018	**0.693±0.009**
20	0.461±0.013	0.558±0.067	0.573±0.027	-	0.598±0.018	**0.717±0.010**
30	0.464±0.015	0.548± 0.048	0.586±0.024	0.502±0.087	0.609±0.019	**0.732±0.011**
40	0.475±0.023	0.547±0.034	0.592± 0.028	0.503±0.060	0.614±0.022	**0.740±0.010**

Table 1: F1 score of HiJoD outperforms all content-based methods on Twitter dataset.

	Twitter			Politifact	
%Labels	CMTF	CMTF++	HiJoD	CMTF	HiJoD
10	0.657±0.009	0.657±0.010	**0.693±0.009**	0.733±0.007	**0.766±0.007**
20	0.681±0.010	0.681±0.009	**0.717± 0.010**	0.752±0.012	**0.791±0.007**
30	0.691±0.010	0.692±0.009	**0.732±0.011**	0.774±0.006	**0.802±0.007**
40	0.699±0.010	0.699±0.009	**0.740±0.010**	0.776±0.004	**0.810±0.008**

Table 2: HiJoD outperforms coupling approaches in terms of F1 score in both datasets.

Testing the effect of different aspects. This experiment refers to the first category of baselines i.e., state-of the-art content-based approaches introduced earlier. Table 1 demonstrates the comparison; label% shows the percentage of known data used for propagation/training of the models. As reported, HiJoD outperforms all content-based approaches significantly. For example, using only 10% of known labels the F1 score of HiJoD is around 12% and 13% percent more than Doc2Vec/SVM and fastText respectively. In case of GloVe/LSTM, due to small size of training set i.e., 10 and 20 percent, the model overfits easily which shows the strength of HiJoD against deep models when there is scarcity of labeled data for training. Moreover, our ensemble model that leverages TTA as one of its embeddings beats the individual decomposition of TTA which illustrates the effectiveness of adding other aspects of the data for modeling the news articles.

Testing the efficacy of fused method in HiJoD. For the second category of baselines, we compare HiJoD against the recent work for joint decomposition of tensors/matrices i.e., CMTF. As discussed earlier, we couple TTA, HTA and TAGS on shared article mode. Since the "term" mode is also shared between TTA and HTA, we also examine the CMTF by coupling on both article and term modes, henceforth referenced as CMTF++ in the experimental results. For the Politifact dataset, there is only one shared mode i.e., article mode. therefore, we only compare against CMTF approach. The experimental results for theses approaches are shown in Table 2. As illustrated, HiJoD leads to higher F1 score which confirms the effectiveness of HiJoD for jointly classification of articles. One major drawback of CMTF approach is that, we have to use a unique decomposition rank for the joint model which may not fit all embeddings and may lead to losing some informative components or adding useless noisy components due to inappropriate rank of decomposition. Another drawback of this technique is that, as we add more embeddings, the optimization problem becomes more and more complicated which may cause the problem become unsolvable and infeasible in terms of time and resources. The time efficiency of HiJoD against CMTF approach is reported in Table. 3; we refer to level-1 and level-2 decompositions as L1D

	L1D		**L2D+Rank finding**		**Total time (Secs.)**	
L1D Rank	CMTF	CP/SVD	CMTF	JIVE	CMTF	HiJoD
5	352.96	7.63	-	13.69	352.96	21.32
10	1086.70	16.35	-	49.69	1086.70	66.04
20	11283.51	44.25	-	133.70	11283.51	177.95
30	13326.80	84.76	-	278.43	13326.80	363.19

Table 3: Comparing execution time (Secs.) of CMTF against HiJoD on Twitter dataset shows that HiJoD is an order of magnitude faster than CMTF approach.

%Labels	Twitter			Politifact		
	JICA	SVD	JIVE	JICA	SVD	JIVE
10	0.684±0.009	0.685±0.017	**0.693±0.009**	0.749±0.006	0.756±0.009	**0.766±0.007**
20	0.710±0.009	0.712±0.014	**0.717±0.010**	0.775±0.006	0.783±0.009	**0.791±0.007**
30	0.724±0.009	0.724±0.018	**0.732±0.011**	0.786±0.006	0.796±0.012	**0.802±0.007**
40	0.734±0.009	0.735±0.012	**0.740±0.010**	0.797±0.006	0.807±0.008	**0.810±0.008**

Table 4: Comparing the efficacy of different approaches for level-2 decomposition illustrates that using JIVE approach for HiJoD ouperforms other methods.

and L2D respectively. As shown, for all ranks, HiJoD is an order of magnitude faster than CMTF due to the simplicity of optimization problem in comparison to equation 7 which means HiJoD is more applicable for real world problems.

Testing the efficacy of using JIVE for level-2 decomposition. In this experiment we want to test the efficacy of JIVE against other approaches for deriving the joint structure of **C**. The evaluation results for this experiment are reported in Table. 4. As shown, SVD and JICA resulted in same classification performance which as explained earlier indicates that the directions of greatest variation and the independent components span the same subspace [29]. However, the F1 score of HiJoD is higher than two other methods on both datasets which practically justifies that considering orthogonal shared and unshared parts and minimizing the residual can improve the performance of naive SVD.

HiJoD vs. single aspect modeling. Finally, we want to investigate how adding other aspects of the articles affect the classification performance. To this end, we decompose our proposed embeddings i.e., TTA, HTA and TAGS extracted from Twitter dataset individually and leverage the **C** matrices for classification. The result of this experiment is demonstrated in Fig. 4. As mentioned before, in contrast to CMTF, in HiJoD we can use different ranks for different aspects as we did so previously. However, in order to conduct a fair comparison between individual embeddings and HiJoD, we merge the embeddings of the same rank. It is worth mentioning that, performance of HiJoD is higher than what is shown in Fig. 4 due to concatenation of **C** matrices of the best rank. So, this result is just for showing the effect of merging different aspects by fixing other parameters i.e., rank and k. As shown, the HiJoD even when we join embeddings of the same rank and do not use the best of which outperforms individual decompositions.

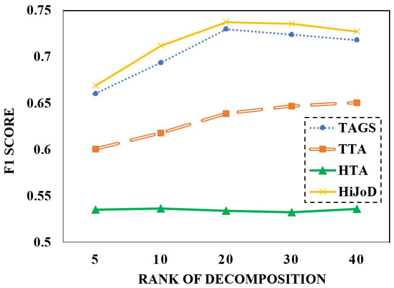

Fig. 4: F1-score of using individual embeddings vs. `HiJoD`. Even when the best rank of each embedding is not used, `HiJoD` outperforms individual decompositions

5 Related Work

Ensemble modeling for misinformation detection. A large number of misinformation detection approaches focus on a single aspect of the data, such as article content [30,2], user features [31], and temporal properties [18]. There exist, however, recent approaches that integrate various aspects of an article in the same model. For example, in [14] the authors propose an ensemble model for finding fake news. In this approach, a bag of words embedding is used to model content-based information, while in this work, we leverage a tensor model i.e., `TTA` which not only enables us to model textual information, but also is able to capture nuanced relations between the words. The different sources of information used in [14] (user-user, user-article and publisher-article interactions) do not overlap with the aspects introduced here (hashtags and HTML features), however, in our experiments we show that `HiJoD` effectively combines both introduced aspects as well as the ones in [14]. In another work [16], news contents and user comments are exploited jointly to detect fake news. Although user comment is a promising aspect, still the main focus is on the words of comments. However, we use HTML tags and hashtags in addition to the textual content.

Semi-supervised Learning / Label Propagation Models. The majority of mono-aspect modeling proposed so far leverage a supervised classifier. For instance, in [7] a logistic regression classifier is used which employs linguistic and semantic features for classification. In [10], authors apply a SVM classifier for content based features. Moreover, some works have been done using recurrent neural network (RNN) and Dynamic Series-Time Structure (DSTS) models [27,23]. In contrary to the aforementioned works, we use a model which achieves very precise classification when leverage very small amount of ground truth. There are some proposed methods that mainly rely on propagation models. For example, in [12] the authors proposed a hierarchical propagation model on a suggested three-layer credibility network. In this work, a hierarchical structure is constructed using event, sub-event and message layers, even though a supervised classifier is required to obtain initial credibility values. In [13], a PageRank-like credibility propagation method is proposed to apply on a network of events, tweets and users. In this work, we leverage belief propagation to address the semi-supervised problem of misinformation detection. We show that proposed approach outperforms state-of-the-art approaches in label scarcity settings.

Acknowledgments

The authors would like to thank Gisel Bastidas for her invaluable help with data collection. Research was supported by a UCR Regents Faculty Fellowship, a gift from Snap Inc., the Department of the Navy, Naval Engineering Education Consortium under award no. N00174-17-1-0005, and the National Science Foundation Grant no. 1901379. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding parties.

6 Conclusion

In this paper, we propose HiJoD, a 2-level decomposition pipeline that integrate different aspects of an article towards more precise discovery of misinformation on the web. Our contribution is two-fold: we introduce novel aspects of articles which we demonstrate to be very effective in classifying misinformative vs. real articles, and we propose a principled way of fusing those aspects leveraging tensor methods. We show that HiJoD is not only able to detect misinformation in a semi-supervised setting even when we use only 10% of the labels but also an order of magnitude faster than similar ensemble approaches in terms of execution time. Experimental results illustrates that HiJoD achieves F1 score of roughly 74% and 81% on Twitter and Politifact datasets respectively which outperforms state-of-the-art content-based and neural network based approaches.

References

1. Acar, E., Levin-Schwartz, Y., Calhoun, V., Adali, T.: Acmtf for fusion of multimodal neuroimaging data and identification of biomarkers (08 2017)
2. Bastidas, G.G., Abdali, S., Shah, N., Papalexakis, E.E.: Semi-supervised content-based detection of misinformation via tensor embeddings. In: ASONAM (2018)
3. Castillo, C., Mendoza, M., Poblete, B.: Information credibility on twitter. In: International Conference on World Wide Web. pp. 675–684. WWW '11 (2011)
4. E. Acar, A. J. Lawaetz, M.A.R., Bro, R.: Structure-revealing data fusion model with applications in metabolomics. In: IEE EMBS (2013)
5. E. Acar, T.K., Dunlavy, D.: All-at-once optimization for coupled matrix and tensor factorizations. In: KDD Workshop on Mining and Learning with Graphs (2011)
6. Gupta, M., Zhao, P., Han, J.: Evaluating Event Credibility on Twitter, pp. 153–164. SIAM (2012)
7. Hardalov, M., Koychev, I., Nakov, P.: In search of credible news. Artificial Intelligence: Methodology, Systems, and Applications. p. 172–180 (2016)
8. Harshman, R.A.: Foundations of the PARAFAC procedure: Models and conditions for an" explanatory" multi-modal factor analysis. UCLA Working Papers in Phonetics **16**(1), 84 (1970)

9. Helmstetter, S., Paulheim, H.: Weakly supervised learning for fake news detection on twitter. In: (ASONAM). pp. 274–277 (2018)
10. Horne, B.D., Adali, S.: This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. CoRR (2017)
11. Hosseinimotlagh, S., Papalexakis, E.E.: Unsupervised content-based identification of fake news articles with tensor decomposition ensembles (2017)
12. Jin, Z., Cao, J., Jiang, Y.G., Zhang, Y.: News credibility evaluation on microblog with a hierarchical propagation model. pp. 230–239. (ICDM) (2014)
13. Jin, Z., Cao, J., Zhang, Y., Luo, J.: News verification by exploiting conflicting social viewpoints in microblogs (2016)
14. K. Shu, A. Sliva, S.W., Liu, H.: Beyond news contents: the role of social context for fake news detection. In: WSDM. pp. 312–320 (2019)
15. K. Shu, L. Cui, S.W.D.L., Liu, H.: Exploiting tri-relationship for fake news detection. arXiv preprint arXiv:1712.07709 (2017)
16. K. Shu, L. Cui, S.W.D.L., Liu, H.: defend: Explainable fake news detection. In: Proceedings of 25th ACM SIGKDD international conference on Knowledge discovery and data mining (2019)
17. Koutra, D., Ke, T.Y., Kang, U., Chau, D., Pao, H.K., Faloutsos, C.: Unifying Guilt-by-Association Approaches: Theorems and Fast Algorithms. In: Machine Learning and Knowledge Discovery in Databases, vol. 6912, pp. 245–260 (2011)
18. Kumar, S., Shah, N.: False information on web and social media: A survey. arXiv preprint arXiv:1804.08559 (2018)
19. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: Bonet, B., Koenig, S. (eds.) AAAI. vol. 333, pp. 2267–2273 (2015)
20. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. ICML 2014 **4** (05 2014)
21. Lin, Z., Feng, M., dos Santos, C.N., Yu, M., Xiang, B., Zhou, B., Bengio, Y.: A structured self-attentive sentence embedding (2017)
22. Lock, E., Hoadley, K., Marron, J., Nobel, A.: Joint and individual variation explained (jive) for integrated analysis of multiple data types. The annals of applied statistics **7**, 523–542 (2013)
23. Ma, J., Gao, W., Wei, Z., Lu, Y., Wong, K.F.: Detect rumors using time series of social context information on microblogging websites. In: CIKM (2015)
24. Papalexakis, E.E., Faloutsos, C., Sidiropoulos, N.D.: Tensors for data mining and data fusion: Models, applications, and scalable algorithms. ACM Trans. Intell. Syst. Technol. **8**, 16:1–16:44 (2016)
25. Papalexakis, Evangelos E: Automatic unsupervised tensor mining with quality assessment. In: SIAM SDM (2016)
26. Rubin, V.L., Conroy, N.J., Chen, Y., Cornwell, S.: Fake news or truth? using satirical cues to detect potentially misleading news (2016)
27. Ruchansky, N., Seo, S., Liu, Y.: CSI: A hybrid deep model for fake news. CoRR
28. Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: A data mining perspective. SIGKDD Explor. Newsl. **19** (2017)
29. Vasilescu, M.A.O.: A Multilinear (Tensor) Algebraic Framework for Computer Graphics, Computer Vision and Machine Learning. Ph.D. thesis, Citeseer (2012)
30. Wu, L., Li, J., Hu, X., Liu, H.: Gleaning wisdom from the past: Early detection of emerging rumors in social media. In: SIAM SDM. pp. 99–107 (2017)
31. Wu, L., Liu, H.: Tracing fake-news footprints: Characterizing social media messages by how they propagate. In: WSDM. pp. 637–645 (2018)