Large-Scale Evaluation of Topic Models and Dimensionality Reduction Methods for 2D Text Spatialization

Daniel Atzberger* , Tim Cech* , Matthias Trapp , Rico Richter ,
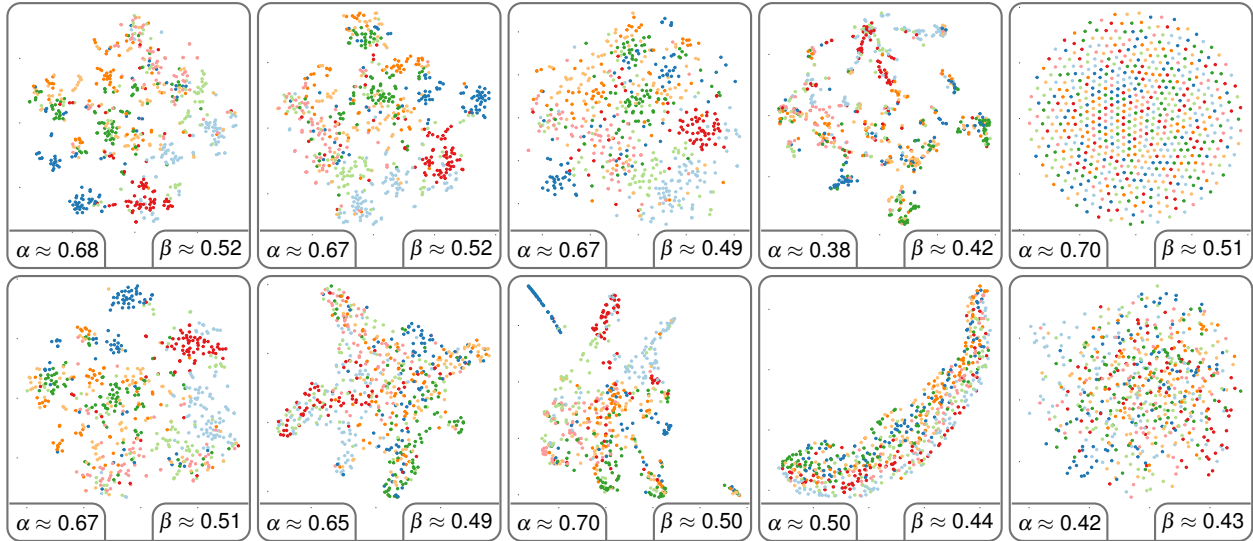Willy Scheibel , Jürgen Döllner , and Tobias Schreck

Fig. 1: Two-dimensional scatter plots derived from topic models and a subsequent dimensionality reduction algorithm. Each point represents the merged source code files of a software project from *GitHub*. The color indicates the associated class of the software project, e.g., *shell*, or *frontend*. The values α and β correspond to our aggregated accuracy and perception metric, respectively.

Abstract—Topic models are a class of unsupervised learning algorithms for detecting the semantic structure within a text corpus. Together with a subsequent dimensionality reduction algorithm, topic models can be used for deriving spatializations for text corpora as two-dimensional scatter plots, reflecting semantic similarity between the documents and supporting corpus analysis. Although the choice of the topic model, the dimensionality reduction, and their underlying hyperparameters significantly impact the resulting layout, it is unknown which particular combinations result in high-quality layouts with respect to accuracy and perception metrics. To investigate the effectiveness of topic models and dimensionality reduction methods for the spatialization of corpora as two-dimensional scatter plots (or basis for landscape-type visualizations), we present a large-scale, benchmark-based computational evaluation. Our evaluation consists of (1) a set of corpora, (2) a set of layout algorithms that are combinations of topic models and dimensionality reductions, and (3) quality metrics for quantifying the resulting layout. The corpora are given as document-term matrices, and each document is assigned to a thematic class. The chosen metrics quantify the preservation of local and global properties and the perceptual effectiveness of the two-dimensional scatter plots. By evaluating the benchmark on a computing cluster, we derived a multivariate dataset with over 45 000 individual layouts and corresponding quality metrics. Based on the results, we propose guidelines for the effective design of text spatializations that are based on topic models and dimensionality reductions. As a main result, we show that interpretable topic models are beneficial for capturing the structure of text corpora. We furthermore recommend the use of t-SNE as a subsequent dimensionality reduction.

Index Terms—Text visualization, spatialization, dimensionality reduction algorithms, topic modeling.

✦

*Both authors contributed equally to this work.

- *Daniel Atzberger, Matthias Trapp, and Willy Scheibel are with University of Potsdam, Digital Engineering Faculty, Hasso Plattner Institute. E-mail: {firstname.lastname}@hpi.uni-potsdam.de*
- *Tim Cech, Rico Richter, and Jürgen Döllner are with University of Potsdam, Digital Engineering Faculty. E-mail: {tcech,rico.richter.1,doellner}@uni-potsdam.de*
- *Tobias Schreck is with Graz University of Technology. E-mail: tobias.schreck@cgv.tugraz.at*

1 INTRODUCTION

Text data is generated in large quantities in numerous application domains, e.g., social media, news articles, scientific articles, and literature. Given a set of documents, a so-called corpus, numerous text visualizations have been proposed to support users in various analytic tasks, e.g., summarization, sentiment analysis, or exploration [42]. In order to "leverage the cognitive benefits from cartography as an established body of knowledge for information visualization", many text visualization techniques rely on a map-like metaphor, i.e., "a map imitation that makes spatialized data appear more like a cartographic map by emphasizing spatial context" [36]. To derive a spatialization, which "encode[s] similarities, [by] mapping each data item to a point on the visual space such that the relative pairwise proximities reflect at best the corresponding pairwise similarities" [55], different *Topic Models* (TMs) and *Dimensionality Reductions* (DRs) are applied for generating two-

Table 1: Examples of text visualizations that rely on a two-dimensional layout derived from a TM and a DR.

Visualization	TM	tf-idf	DR	linear combination
Skupin [66]	VSM	✗	SOM	✗
Kuhn et al. [44]	LSI	✗	MDS	✗
Fried et al. [31]	LSI	✓	MDS	✗
Caillou et al. [13]	LSI	✓	UMAP	✗
Kim et al. [40]	NMF	✗	t-SNE	✗
Choo et al. [17]	NMF	✗	t-SNE	✗
Linstead et al. [47]	LDA	✗	MDS	✗
Gansner et al. [33]	LDA	✗	MDS	✗
Atzberger et al. [7, 8]	LDA	✗	MDS	✓
Kucher et al. [43]	LDA	✗	t-SNE	✗
Yan et al. [80]	LDA	✗	t-SNE	✗

dimensional layouts for a corpus. Starting from the *Document-Term Matrix* (DTM) representation of a corpus, which stores the frequency of the terms within each document, TMs aim to detect clusters within the vocabulary, so-called topics, by analyzing patterns of co-occurring words [19]. Thereby, TMs are specified by several hyperparameters that control the underlying model and the training algorithm. As a result, TMs yield a high-dimensional vector representation of the documents, which are further projected to a lower-dimensional space by a DR for visualization. Table 1 lists examples of text visualization techniques using a two-dimensional spatialization derived from a TM and a DR. To summarize, starting from a high-dimensional description of the documents within a corpus that stores the term frequencies, combining a topic model and a subsequent dimensionality reduction results in a two-dimensional scatter plot, where each point represents a single document.

A visualization designer has to select a TM and a DR and specify the corresponding hyperparameters. Figure 1 shows exemplarily that the concrete choice of the combination and assignment of the hyperparameters strongly influences the resulting layout regarding clustering and global structure. TMs and DRs are often treated as a "black box" without thoroughly investigating the effect of the hyperparameters. In most cases, publications presenting a text visualization focus on the visual mapping, whereas a comparison of several possible layouts is usually omitted. Other studies that focus on deriving guidelines for the effective use of DRs for visualization tasks have not yet considered TMs. We close this gap by investigating the influence of TMs and DRs on the spatialization of text visualizations. The research space of TM, DR, and test data set is extensive. We narrow down this large experimental space by means of a computing cluster and a representative selection of relevant methods. This is, to the best of our knowledge, the most comprehensible experimental evaluation in this area.

We present a benchmark $\mathcal{B} = (\mathcal{D}, \mathcal{L}, \mathcal{Q})$ composed of a set of corpora $\mathcal{D}$, layout algorithms $\mathcal{L}$, and quality metrics $\mathcal{Q}$. The set $\mathcal{D}$ contains five corpora, each given by a DTM, with assigned class labels for each document. Each layout in $\mathcal{L}$ emerges from a combination of a TM and a subsequent DR. The set $\mathcal{L}$ comprises 52 different layout algorithms composed of 13 TMs and 4 DRs. The metrics in $\mathcal{Q}$ quantify the preservation of local and global structures of the high-dimensional corpora in the two-dimensional layout and their cluster separation. We quantify each layout and the influence of their hyperparameters by performing a grid search using a computational cluster. We derived a tabular dataset from the execution of the benchmark with more than 45 000 sampled layouts and their quality scores. Based on the dataset, we show the performances with respect to accuracy and the effectiveness to perceive clusters and propose guidelines for the effective use of TMs and DRs for the spatialization of corpora. To summarize, we make the following contributions:

1. We provide a benchmark of five corpora, 52 layout algorithms, and eight quality metrics for evaluating the use of TMs and DRs for two-dimensional text spatialization. We provide our imple-

mentation as a Git repository[1].

2. We generate a large multivariate dataset with more than 45 000 entries containing the quality score of different hyperparameter configurations. We selected the range of the hyperparameters following the recommendations of the used libraries.

3. We provide high-level insights into the aggregated performances and propose guidelines for the practical use of TMs and DRs.

The remaining part is structured as follows: we present related work in Section 2. The structure and composition of our benchmark, together with technical details on the execution, are detailed in Section 3. In Section 4, we analyze the resulting multivariate dataset. We discuss the results and present guidelines together with threats to validity in Section 5. We conclude this work and present directions for future work in Section 6.

2 RELATED WORK

Several text analysis tasks, e.g., text classification [2], text summarization [54], or text clustering [3], rely on TMs for modeling documents within a corpus. In most cases, TMs are evaluated using statistical measures, e.g., perplexity or coherence scores [60], or by asking users about the interpretability of the derived topics [48]. Although TMs are often part of text visualizations or intended to be represented by a visualization, e.g., for topic comparison [5, 64, 79], for topic evolution [21], or corpus exploration [25, 58], in most cases TMs are not evaluated. An exception is the work of Riehmann et al., who showed in an expert study that the topics extracted by LDA do not match the list that experts curated [59]. DRs are among the popular techniques for visualizing high-dimensional data, e.g., by computing a set of scatter plots [46], or by computing two-dimensional layouts as shown in Table 1. Thereby, DRs are widely evaluated and discussed in previous surveys with differing foci: surveys that focus on the mathematical principles of DRs and quantitative studies. In the latter, either the accuracy of DRs, i.e., the preservation of local and global structures, or the effectiveness of the resulting layout for human perception are investigated.

2.1 Mathematical Surveys of Dimensionality Reductions

One of the earlier surveys on DRs with a mathematical introduction was presented by Fodor [30]. Using a similar approach, Cunningham and Ghahramani discussed several linear DRs [20]. Further, Engel et al. presented a survey of basic DRs from a visualization point of view [26]. In addition to the theoretical alignment of the individual DRs, the authors also compare their underlying assumptions, online compatibility, and computational cost. Nonato and Aupetit recently presented a comprehensive survey of DRs specific to the visualization domain [55]. In addition to a detailed taxonomy of DRs, distortion types, analytics tasks, and layout enrichment methods, the authors formulate guidelines on choosing a DR for a given analytics task. Unlike our work, the authors additionaly formulate guidelines derived from their proposed taxonomy and the constraints on mathematical properties of the DRs, rather than from experimental results.

2.2 Evaluating Dimensionality Reductions for Accuracy

One method for assessing visualization techniques is to define aspects of quality and derive quality metrics for quantitative measurements [11]. To derive guidelines for the effective use of DRs for visualization techniques, benchmarks that evaluate a large number of different layouts have been proposed. One quality aspect for DRs is the accuracy, i.e., the preservation of local and global structures from the high-dimensional dataset in the low-dimensional representation, for which different metrics have been developed [53]. For example, van der Maaten et al. proposed a benchmark for comparing different DRs [68]. Their study compared the quality of twelve non-linear methods with Principal Component Analysis (PCA) on ten different datasets by measuring three quality metrics. As a main result, the authors showed that the non-linear methods did not outperform PCA. However, t-SNE and UMAP were not included in the benchmark. One study by Gisbrecht and Hammer

---

[1] hpicgs/Topic-Models-and-Dimensionality-Reduction-Benchmark

Table 2: Characteristics for the five datasets in our benchmark containing the number of documents m, the median size of the documents N, the size of the vocabulary n and the number of categories k. The size represents the size of the raw dataset given by $m \cdot n \cdot 8$ byte. This measurement neglects program overhead and memory required by the DR and TM.

Dataset	Size	m	N	n	k
20 Newsgroup	575.9 MiB	11 314	176	6 672	20
Emails	486.0 MiB	9 111	182	6 992	4
GitHub Projects	2 024.5 MiB	653	52 635	405 117	8
Reuters	205.5 MiB	9 122	102	2 953	65
Seven Categories	993.8 MiB	3 127	396	11 373	7

presents the mathematical principles of non-linear DRs and an evaluation of their performance on three datasets [35]. In particular, the authors investigate the influence of individual hyperparameters on the quality of the resulting layouts. Recently, Espadoto et al. presented an architecture for a DR benchmark to support users in selecting appropriate DRs and allow researchers to assess new methods [29]. Further, the authors evaluated the benchmark to derive insights and best practices for the effective use of DRs for visualization tasks [28]. Their benchmark comprises 18 datasets, 44 dimensionality reductions, and seven quality metrics. Although the authors consider text data as one of three types besides tabular and image data, their benchmark does not consider TMs. Similarly, Vernier et al. followed this approach and investigated which DRs are suitable for visualizing dynamic data [71]. For this purpose, the benchmark was extended by metrics that measure the temporal stability of the generated layouts. Until then, the assessment of temporal stability was approached by human judgement [34]. In a later work, Vernier et al. extended their benchmark by two versions of t-SNE that improved spatial quality and temporal stability [70].

2.3 Evaluating Dimensionality Reductions for Perception

Complimentary to accuracy metrics, perception metrics quantify the effectiveness of a two-dimensional layout for perceiving structures, e.g., class separation [4, 62]. Morariu et al. presented a benchmark for investigating how quality metrics are suitable to describe the visual appearance of two-dimensional layouts derived from DRs [53]. For this purpose, rankings provided by study participants were used as labels to predict user preferences given quality metrics. Similarly, Xia et al. presented a convolutional neural network for modeling the human perception of visual clusters [77]. Their model is trained on a human-labeled dataset and a qualitative study determining influence factors for cluster perception. Using a similar approach, Wang et al. combined quantitative measurements and human judgments to evaluate their proposed perception-driven DR to maximize the perceived class separation [73]. Xia et al. presented a contrastive DR approach that considers accuracy metrics in addition to optimizing visual cluster separation by measuring three perception metrics [76]. In addition to a quantitative assessment through metrics, Xia et al. showed, through a user study, that their approach outperformed t-SNE and UMAP in the task of cluster identification. A further work that relies on a user study was presented by Sedlmair et al. who investigated to what extent 3D scatter plots or scatter plot matrices improve the perception of cluster separation, compared to 2D scatter plots [63]. Similarly, Xia et al. conducted a user study to investigate which DRs are suitable for visual cluster analysis tasks, e.g., cluster identification, membership identification, distance comparison, and density comparison [78].

3 BENCHMARK $\mathcal{B}$

Neither the surveys nor the evaluations of the existing benchmarks consider TMs as components of the layout process. Even for DRs developed for visualizing text corpora, TMs were not considered in their evaluation [16, 37]. Our work addresses this gap by evaluating a benchmark that explicitly considers text corpora and TMs as essential layout components. The idea of such a benchmark was previously proposed by Atzberger and Cech et al. [9]. The authors proposed a benchmark $\mathcal{B} = (\mathcal{D}, \mathcal{L}, \mathcal{Q})$, consisting of a set of text corpora $\mathcal{D}$,

a set of layouts $\mathcal{L}$, and a set of quality metrics $\mathcal{Q}$. We revised and extended this approach with respect to the following three aspects: (1) we only consider corpora in $\mathcal{D}$, where the given categories correspond to semantic concepts. (2) Our layouts $\mathcal{L}$ contain additional TMs but focus on a subset of the proposed DRs. (3) In addition to accuracy metrics, our set of quality metrics $\mathcal{Q}$ further contains measures for quantifying the perceptual effectiveness of the resulting layout.

3.1 Datasets $\mathcal{D}$

Our set $\mathcal{D}$ contains five corpora, whose characteristics are summarized in Table 2. Each element $D = (C, P) \in \mathcal{D}$ is given as a pair consisting of a corpus $C = \{d_1, \ldots, d_m\}$ and a partition $P = \{c_1, \ldots, c_k\}$, i.e., a disjoint decomposition of C in k classes. The corpus C consisting of m documents $d_1, \ldots, d_m$ over a vocabulary $\mathcal{V}_D$ of size $n = |\mathcal{V}_D|$ is given as a DTM, in which the entry in cell (i, j) indicates the frequency of the term $w_j \in \mathcal{V}_D$ in document d_i. The elements in a class $c \in P$ are documents that belong to a higher-level concept.

Four of the five corpora in $\mathcal{D}$ are preprocessed versions of the commonly used datasets *20 Newsgroup*[2], *Emails*[3], *Reuters*[4], and *Seven Categories*[5]. We applied standard preprocessing steps, e.g., removal of stop words and the lemmatization of the vocabulary, and additional, dataset-dependent steps, e.g., removing the email header in the 20 Newsgroup dataset. For details, we refer to our Git repository. Our fifth dataset covers the domain of software visualization. Previous work has shown that source code is suitable for text analysis by TMs [10, 15]. As one corpus on source code, we propose a *GitHub Projects* corpus containing 653 documents from eight categories, i.e., where each document contains the merged source code files of a software project that belongs to one particular *GitHub topic*[6]. Based on the file extension, we detect source code files written in one of the following languages: *C*, *C++*, *C#*, *Go*, *Java*, *JavaScript*, *Move*, *PHP*, *Python*, *Ruby*, *Rust*, or *Solidity*. Thereby, we collected the 100 most popular projects ranked by stars[7] for each of the following GitHub topics: *cryptocurrency*, *data-visualization*, *machine-learning*, *frontend*, *database*, *shell*, *server*, and *3d*. As we assume a disjoint partitioning, we consider only the first mention of a project within the query results. Most remarkable is the large size of the vocabulary n as shown in Table 2. This is because, before preprocessing, terms are included, such as short identifier names, which do not occur in English. In addition to the usual preprocessing, source code-specific operations are performed, such as separating identifier names according to standard naming conventions and filtering keywords of programming languages as they carry no semantic meaning. Ideally, after preprocessing, the vocabulary contains all English language words that occur as comments or identifiers. However, whether the term originates from a comment or an identifier name is not distinguished. This procedure is typical for the application of topic models to source code [15].

3.2 Layouts $\mathcal{L}$

The elements of the set of layouts $\mathcal{L}$ originate from combinations of a TM and a subsequent DR. Training a TM on a given corpus, given by a DTM, yields a document representation in a Euclidean standard space of dimensionality $\leq n$. The high-dimensional document representations are projected using a DR on a two-dimensional plane. Particular combinations applied in existing visualization approaches are summarized in Table 1.

3.2.1 Topic Models

From the DTM description of a corpus, each document is given as an n-dimensional vector containing the absolute frequencies of each term. Together with a similarity measure between documents, this

[2] scikit-learn.org/0.19/datasets/twenty_newsgroups.html
[3] kaggle.com/datasets/dipankarsrirag/topic-modelling-on-emails
[4] kaggle.com/datasets/nltkdata/reuters
[5] kaggle.com/datasets/deepak711/4-subject-data-text-classification
[6] github.com/topics
[7] docs.github.com/en/get-started/exploring-projects-on-github/saving-repositories-with-stars

representation is denoted as the *Vector Space Model* (VSM) [19]. In our considerations, we use the cosine similarity for documents, as it allows the comparison of documents of different lengths. However, the DTM only contains the absolute frequencies of a term regardless of whether the term is also frequently represented in other documents. In practice, however, the terms represented in only a few documents often indicate an underlying concept and are particularly relevant. By weighting the entries in the DTM according to the *term frequency-inverse document frequency* (tf-idf) scheme, the VSM can be modified to incorporate this effect [3]. Specifically, the tf-idf of a term w in document $d \in C$ is given by the product of the term-frequency of the term w in d and the inverse-document-frequency of w in d, i.e.,

$$\text{tf-idf}(w,d) = \frac{n(w,d)}{\sum\limits_{d' \in C} n(w,d')} \cdot \log\left(\frac{|C|}{|\{d' \in C | w \in d'\}|}\right), \qquad (1)$$

where $n(w,d)$ denotes the frequency of term w in document d.

The DTM is a sparse matrix, i.e., most entries are zero, as documents usually contain a small fraction of the entire vocabulary. Most TMs aim to find a more compressed representation of the DTM by grouping co-occurring words into topics. Algorithms that detect topics as part of their results are called topic models. Topics are thereby given as vectors of size n, with the ith entry containing a weight that describes the impact of term w_i for the topic. From the most relevant words, a human-interpretable concept can be inferred in most cases. In that sense, the VSM and its tf-idf weighted variant are not a topic model. In our considerations, we will cover four different topic models described below. However, in the following, when talking about the evaluation of all TMs, we also include the VSM and its tf-idf weighted variant. For example, *Latent Semantic Indexing* (LSI) is based on Singular Value Decomposition (SVD), which results in a decomposition of a given DTM into a document-topic matrix and a topic-term matrix [23]. In practice, tf-idf weighting is often applied to increase the interpretability of the topics. Another linear algebra approach for topic modeling is *Non-Negative Matrix Factorization* (NMF), where the DTM or its tf-idf-weighted variant is approximated as a product of two matrices, i.e., a document-topic matrix and a topic-term matrix [45]. *Latent Dirichlet Allocation* (LDA) is a probabilistic approach for topic modeling and is probably the most widely used TM in the visualization domain. LDA is based on the assumption of a generative process underlying a corpus. Training an LDA model results in topics that are given as multinomial distributions over the vocabulary. Further, each document is represented as a multinomial distribution over the topics [12]. As documents are given as distributions, we specifically apply the Jensen-Shannon distance for measuring the similarity between documents. As LDA is a probabilistic model, we do not replace the DTM with the tf-idf weighted entries. As the last TM, we integrated *Bidirectional Encoder Representations from Transformers* (BERT), which is a deep learning-based approach for topic modeling that is known to generate easily interpretable topics [24]. Unlike the other TMs, each document is described as a high-dimensional vector associated with exactly one or zero topics. The topics are then derived from these associations using a class-based tf-idf weighting. In the case of BERT, the similarity between documents is again given by the cosine similarity. According to our survey of works, these methods are representative of topic extraction in a significant part of the document visualization literature.

3.2.2 Dimensionality Reductions

As DRs, we consider t-SNE and UMAP, as they have shown promising results in earlier studies [28]. We further consider MDS and SOMs, as they are widely used in the text visualization domain, as shown in Table 1. Although many more dimension reductions exist, we limit our considerations to these four for capacity reasons. *t-distributed Stochastic Neighbor Embedding* (t-SNE) is a DR designed to preserve local structures within a dataset [67]. This is accomplished by assuming a Gaussian distribution centered around each point in the given high-dimensional space, representing the probability of picking another point as a neighbor. The number of effective neighbors considered is controlled by the perplexity hyperparameter, which allows to trade off local and global properties. The main goal of t-SNE is the preservation of neighborhoods in the low-dimensional representation with respect to a t-distribution. The final layout is obtained by an iterative optimization process that minimizes a stress function that measures the difference in overall similarity scores derived from the respective distributions. *Uniform Manifold Approximation and Projection* (UMAP) was developed to address the shortcomings of t-SNE, e.g., the distances between clusters in a t-SNE plot allow no interpretation [50]. Conceptually similar to t-SNE, UMAP differs in its mathematical details, e.g., it relies on a stress function derived from Cross-Entropy rather than the Kullback-Leibler divergence. UMAP has two hyperparameters: the number of neighbors as a trade-off between preserving local and global structures and the minimal distance that controls how close data points can be grouped together in the two-dimensional layout. *(Metric) Multi-dimensional Scaling* (MDS) operates on a dissimilarity matrix of the dataset, i.e., a matrix that contains the pairwise distances between the data points. MDS aims to compute a lower-dimensional representation, such that the pairwise Euclidean distances between the points in the layout reflect the entries in the dissimilarity matrix [18]. In particular, MDS allows for the visualization of abstract datasets that are not embedded in the Euclidean space. The positions of the data points are computed iteratively by optimizing a stress function, for example, using the SMACOF algorithm. The number of iterations is the only hyperparameter of the model. *Self-Organizing Maps* (SOMs) are a class of fully-connected two-layered neural networks where the neurons of the second layer are arranged on a two-dimensional grid, whose width and height are given by two hyperparameters [41]. For a given input, the neuron whose weight vector is most similar to the input is activated. This so-called best matching unit determines the position of the given input vector in the two-dimensional space. The weights are adjusted during the training phase starting from random initialization in order to minimize the sum of all quantization errors, i.e., the differences between the input vectors and their best matching unit. In the case of the SOM, we applied a PCA that captures $95\,\%$ of the variance for a given dataset to reduce computational efforts [39].

The combination of a TM with a DR allows for special considerations. For example, in the case of LDA, the similarities between the topics differ, as they are given by multinomial distributions over the vocabulary. Applying a DR on the document-topic representation does not consider those similarities, thus treating the topics as orthogonal to each other. Atzberger et al. proposed an alternative by first applying the DR on the topics and then aggregating the positions of the documents as linear combinations according to their document-topic representation [7]. The position $\bar{d}$ of document d is therefore given by

$$\bar{d} = \sum_{j=1}^{K} \theta_j \bar{\phi}_j, \qquad (2)$$

where $\theta = (\theta_1, \ldots, \theta_K)$ denotes the topic representation of d, and $\bar{\phi}_1, \ldots, \bar{\phi}_K$ denotes the positions of the topics after application of a DR.

3.3 Quality Metrics $\mathcal{Q}$

The elements in the set $\mathcal{Q}$ are quality measures that quantify certain layout aspects. Similar to previous benchmark studies, we measure the quality of a layout with respect to the local and global structures of a corpus by using several accuracy metrics. The *Trustworthiness* α_T measures the percentage of close points in the two-dimensional layout that are also close in the VSM [69]. Vice versa, the *Continuity* α_C measures the percentage of points in the VSM that are also close in the two-dimensional layout [69]. For both metrics, we refer to the seven nearest neighbors, as suggested in previous studies [28, 70, 71]. The *7-Neighborhood hit* α_{NH} requires labels for each document. It measures the percentage of points with the same label among the seven nearest neighbors, averaged over all points [57]. All three metrics have values in the $[0,1]$ range, with 1 being the optimal score. The last accuracy metric is based on the *Shephard Diagram*, a two-dimensional scatter plot that relates the pairwise distances in D to the Euclidean distances

Table 3: Range for the hyperparameters considered in our experiments. Each configuration for one DR is combined with a dataset and TM.

DR	Parameter Name	Values
t-SNE	learning_rate	250, 1000, 2000, 4000, 10000
t-SNE	n_iter	10, 17, 28, 46, 77, 129, 215, 359, 599, 1000
t-SNE	perplexity	5–50 step size 5
UMAP	min_dist	0.0–1.0 step size 0.1
UMAP	n_neighbors	2, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20
SOM	m	10–20 step size 1
SOM	n	10–20 step size 1
MDS	max_iter	300–900 step size 20

in the layout [38]. In an ideal scenario, the Shephard Diagram would be a subset of the diagonal. Then, the *Shephard Digram Correlation* α_{SDC} is a measure for the deviation from the ideal scenario. It is given by the Spearman Rank Correlation of the Shephard Diagram. The metric ranges between $[-1, 1]$, with 1 being the optimal score.

As a second group of metrics, we approximate the effectiveness of perceiving resulting clusters and existing categories. Following the results of Sedlmair and Aupetit, we include the *Distance Consistency* β_{DC} [62]. It measures the percentage of points in the projected two-dimensional space whose category center, i.e., the average of all points in that category, is also its nearest category center [65]. The *Silhouette Coefficient* β_{SC} compares the mean intra-cluster distance a and the mean inter-cluster distance b, where in our case, the cluster labels are given by the categories [61]. By dividing the difference $b - a$ by $\max(a, b)$, the Silhouette Coefficient ranges between $[-1, 1]$, with 1 being the optimal score. We further apply the *Calinski-Harabasz index* β_{CH} [14] and the *Davies-Bouldin-index DB* [22]. In both cases, the metric results in a non-negative value. In our benchmark, we normalize both by dividing them by the maximal value achieved on a dataset-TM combination, with 1 being the optimal score for β_{CH} and 0 being the optimal score for β_{DB}.

3.4 Experimental Setup & Implementation

We implemented this benchmark using Python 3 and state-of-the-art libraries for all TMs and DRs. Regarding deployment, the implementation was designed for concurrent execution on a computational cluster. The source code of the benchmark, including the scripts to generate the GitHub Projects dataset, is available in a Git repository[1].

3.4.1 Hyperparameter Settings

For each dataset, we consider precisely one version of each TM, using a fixed hyperparameter configuration chosen using best practices [72]. In the case of just a few categories k, i.e., for the Emails corpus, the Seven Categories corpus, and the GitHub Projects corpus, we set the number of topics $K = 2k$. For the 20 Newsgroup corpus and the Reuters corpus, we set $K = k$. As a general check for plausibility, we further inspected the most relevant words for each topic with respect to interpretability. We limited the benchmark regarding the TMs as iterating over the hyperparameters of each TM, e.g., the number of topics, or the Dirichlet priors of the LDA model, would enlarge the benchmark by multiple orders of magnitude. The ranges for the hyperparameters of the four different DRs are summarized in Table 3. In principle, the validation of the individual hyperparameters follows a grid search, resulting in above 45 000 different hyperparameter configurations. However, the order of the layout computation is random to ensure that representative results are available during preliminary analysis.

3.4.2 Software Dependencies

Our implementation is based on Python 3.10 and several actively maintained and widely used third-party libraries. For LDA, LSI, and NMF, we have chosen the *Gensim* library (4.2.0), and for BERT, the *Sentence Transformer* library (2.2.2). As the application of BERT relies on pretrained word embeddings, it is a deterministic approach to topic modeling. The algorithms provided by the Gensim implementation are

also deterministic after initialization. We have chosen the *distilbert-base-nli-mean-tokens* as Sentence Transformer[8]. For t-SNE and MDS, we use *Scikit-Learn* (1.2.1) for the Silhouette Coefficient, the Calinski-Harabasz index, and the Davies-Bouldin index as well. For UMAP, we use the *UMAP-Learn* library (0.5.3). For the SOM, we have chosen the implementation provided by *Sparse-SOM* (0.6.1) [51]. Furthermore, our preprocessing is based on the libraries *NLTK* (3.7), e.g., for removal of stop words, and *Spacy* (3.4.3) for lemmatization.

3.4.3 Computational Cluster

We set up the benchmark on a computational cluster with *Simple Linux Utility for Resource Management* (SLURM) [81] for concurrent execution. This cluster allowed for a large speed-up while requiring special handling for the software deployment and job scheduling. We had access to the AMD x64 nodes, which came in two kinds of hardware configurations: (1) HPE XL225n Gen10 machines with 2 AMD EPYC 7742 processors, 512 GiB RAM, and 64 cores, and (2) Fujitsu RX2530 M5 machines with 2 Intel Xeon Gold 5220S processors, 96 GiB RAM, and 32 cores. From the HPE XL225n Gen10 nodes, we regularly used 11, and from the Fujitsu RX2530 M5, we regularly used 10. The SLURM setup on each node required to use an Enroot[9] container with the benchmark. We derived this Enroot container from a Docker[10] container using Pyxis[11] and reused the container via caching.

Scheduling a job via SLURM requires explicit specification of required resources, e.g., RAM, as upper limits. Given our heterogeneous dataset sizes (Table 2), we decided to split up job allocations into two running sets, a memory-heavy and a memory-moderate one. The memory-heavy set covers the evaluation for the GitHub Projects dataset with a RAM allocation of 200 GiB. The memory-moderate set covers the other datasets with a RAM allocation of 40 GiB. Overall, using this design, we ran thousands of jobs corresponding to ten thousands of different layouts on the cluster. On average, a job of the memory-heavy set had an execution time of 12 hours with a maximum of 36 hours. For the memory-moderate set, the average execution time was around 2 hours, with a maximum of 10 hours.

4 RESULTS

We evaluated 46 311 samples, which corresponds to $\approx 94.7\%$ of targeted layouts. The remaining 5.3% of targeted layouts could not be computed due to exceeding memory consumption. The corresponding quality metrics are stored as a tabular dataset. Further, the dataset is augmented with two additional aggregated quality metrics concerning accuracy and perception. The aggregated quality metrics are given as linear combinations of the accuracy and perception metrics, respectively, taking into account a correlation analysis to limit the influence of strongly correlated quality metrics. We then performed an analysis of this dataset with respect to four specific questions:

1. Do the tf-idf weighting scheme and Equation (2) improve the results?

2. Which layout achieves the best result for a given dataset with respect to accuracy or perception?

3. How sensitive are the DRs with respect to their hyperparameters?

4. What is the performance of the default hyperparameters?

4.1 Correlation of Quality Metrics

Previous ad-hoc formulations of aggregated metrics relied on the arithmetic mean of the individual metrics, i.e., each metric is weighted equally [28,53,70,71]. Instead, we target a correlation-adjusted weighting that groups strongly correlated quality metrics using a threshold of 0.8. Specific to the correlation analysis, the David-Bouldin index β_{DB} has been replaced by $1 - \beta_{DB}$ to achieve its optimal score at 1. The pairwise correlations are shown in Figure 2. For the accuracy metrics, we

[8]metatext.io/models/sentence-transformers-distilbert-base-nli-stsb-mean-tokens

[9] ⚙ NVIDIA/enroot

[10]docker.com

[11] ⚙ NVIDIA/pyxis

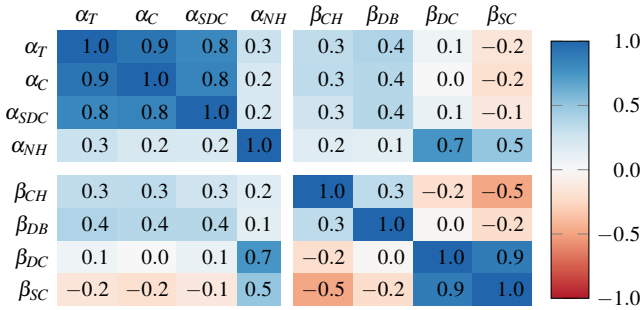|	α_T	α_C	α_{SDC}	α_{NH}	β_{CH}	β_{DB}	β_{DC}	β_{SC}
α_T	1.0	0.9	0.8	0.3	0.3	0.4	0.1	−0.2
α_C	0.9	1.0	0.8	0.2	0.3	0.4	0.0	−0.2
α_{SDC}	0.8	0.8	1.0	0.2	0.3	0.4	0.1	−0.1
α_{NH}	0.3	0.2	0.2	1.0	0.2	0.1	0.7	0.5
β_{CH}	0.3	0.3	0.3	0.2	1.0	0.3	−0.2	−0.5
β_{DB}	0.4	0.4	0.4	0.1	0.3	1.0	0.0	−0.2
β_{DC}	0.1	0.0	0.1	0.7	−0.2	0.0	1.0	0.9
β_{SC}	−0.2	−0.2	−0.1	0.5	−0.5	−0.2	0.9	1.0

Fig. 2: Heatmap showing the pairwise correlations between the eight quality metrics using a diverging color scheme.

observe that the three metrics α_T, α_C, and α_{SDC} are strongly positively correlated. We, therefore, merge them into a single quality metric using their average. Conversely, the three metrics correlate weakly with α_{NH}. Overall, we define the aggregated accuracy metric α as

$$\alpha = \frac{1}{2}\alpha_{NH} + \frac{1}{2}\left(\frac{\alpha_T + \alpha_C + 0.5 \cdot (\alpha_{SDC} + 1)}{3}\right), \quad (3)$$

where we replace the Shephard Diagram Correlation α_{SDC} by $0.5 \cdot (\alpha_{SDC} + 1)$ such that it ranges between 0 and 1. Our metric α is in the value range $[0, 1]$, with 1 being the optimal score. Regarding the perception metrics, the only strong correlation occurred between β_{DC} and β_{SC}. The other pairwise correlations do not allow for further grouping. The aggregated perception metric β is defined as

$$\beta = \frac{1}{3}(1 - \beta_{DB}) + \frac{1}{3}\beta_{CH} + \frac{1}{3}\left(\frac{0.5 \cdot (\beta_{SC} + 1) + \beta_{DC}}{2}\right). \quad (4)$$

As before, the single metrics are modified such that the aggregated metric β ranges between $[0, 1]$ with 1 being the optimal score.

In our benchmark, we selected datasets with predefined categories. As the predefined labels indicate a "higher-level" concept, i.e., semantic themes within the documents, we assume that such a higher-level concept shows relations to the vocabulary, and therefore a TM would yield topics that can be associated with the predefined categories. A good layout algorithm with respect to the accuracy metric α would result in a two-dimensional scatter plot so that documents that share the same "dominant" topic form a cluster. Concerning the perception metric, these clusters should be separated well. Our scatter plots of the best results support this conjecture.

4.2 Binary Decisions

The choice of weighting the VSM according to the tf-idf scheme is binary. Using a binary test[12], we investigate whether the tf-idf weighting improves the accuracy metric α and the perception metric β. We extract pairs of layouts from our result dataset where the tf-idf scheme has been applied in one case but not the other, while all the other hyperparameters are the same. The number of pairs in which the tf-idf improves the results is denoted as k. The p-values and the lower bounds for the confidence intervals (the upper bound is always one since our unknown parameter is the probability that the tf-idf weighting improves the result) for the confidence level of 0.99 are shown in Table 4. The p-values of the complete set of pairs, i.e., named Total in Table 4, show that the layout algorithms significantly improve the results concerning α and β. However, different results might occur on selected datasets, e.g., in the case of the 7 Categories dataset concerning β.

Also, the application of Equation (2) is a binary choice. Analogously to before, we applied a binary test. The results are shown in Table 5. From the p-values of the complete set of pairs, i.e., the total case in Table 5, we conclude that Equation (2) improves the results concerning accuracy and perception.

---

[12] docs.scipy.org/doc/scipy/reference/generated/scipy.stats.binomtest.html

Table 4: Results of the binary test for the null hypothesis "The tf-idf weighting scheme does not improve the results according α, or β respectively." As the p-values (Total) are 0.00, we reject the zero hypotheses.

Dataset	n	Accuracy Metric α			Perception Metric β		
		k	p-value	Conf.	k	p-value	Conf.
20 Newsgroup	2842	2577	0.00	0.89	2839	0.00	1.00
7 Categories	2897	1894	0.00	0.63	655	1.00	0.21
Emails	2909	2667	0.00	0.90	2516	0.00	0.85
GitHub	2629	2344	0.00	0.88	2370	0.00	0.89
Reuters	2895	1454	0.41	0.48	1450	0.47	0.48
Total	**14172**	**10936**	**0.00**	**0.76**	**9830**	**0.00**	**0.68**

Table 5: Results of the binary test for the null hypothesis "Equation (2) does not improve the results according α, or β respectively." As the p-values (Total) are 0.00, we reject the zero hypotheses.

Dataset	n	Accuracy Metric α			Perception Metric β		
		k	p-value	Conf.	k	p-value	Conf.
20 Newsgroup	3594	2404	0.00	0.65	1734	0.98	0.46
7 Categories	3651	2117	0.00	0.56	2057	0.00	0.54
Emails	3660	2122	0.00	0.56	1753	0.99	0.46
GitHub	3124	1898	0.00	0.59	1745	0.00	0.54
Reuters	3646	2532	0.00	0.68	1841	0.28	0.49
Total	**17675**	**11073**	**0.00**	**0.62**	**9130**	**0.00**	**0.51**

4.3 Optimal Results

We first consider the optimal values for α and β for each layout on each dataset. The results are summarized in Figure 3. Some layouts could not be computed (grey cells) due to exceeding memory consumption. For example, BERT is based on a pre-trained embedding, which has not been shown suitable for modeling source code. This is because the use of identifiers in source code differs from natural language as identifiers without vocals might have a semantic meaning, e.g., "ccxt" has no vocal and is, therefore, no "natural" word but refers to a cryptocurrency trading API that is well known among practitioners.

We assigned each layout algorithm an identifier given as a quadruple. The first entry indicates the TM. The second entry indicates whether the tf-idf weighting was applied (+), or not (-), or could not be considered for the specific TM (X). The third entry contains the DR. The fourth entry indicates whether the position of the documents was computed according to Equation (2) as a linear combination (+), or not (-), or could not be considered, as no topics were extracted (X).

4.3.1 What Layouts perform best for a given dataset?

The optimal results for each dataset are summarized in Table 6. No layout algorithm scores best over all datasets with respect to α. In any case, the first two entries of the quadruple are either given by (LSI,+), (LSI,-), or (LDA, X). We suspect the reduction in dimensionality within LSI and LDA to be an advantage for a subsequent DR compared to the VSM and BERT. The last two entries specifying the best DR are either given by (t-SNE,+), (t-SNE,-), or (UMAP,+). This observation is aligned with the results of Espadoto et al. in the case of the VSM [28]. However, most of the layout algorithms perform similarly. Only MDS shows a generally lower performance compared to the other DRs. The optimal results with respect to β are summarized in Table 7. As before, the best results are achieved when t-SNE is applied. However, contrary to our observations on α, LDA and LSI perform inferior to the baseline VSM approach. As our metric β mainly quantifies the cluster separation, this observation confirms the prevailing opinion that t-SNE produces well-separated clusters.

4.3.2 What is the influence of the parameters n, m, k?

Regarding α, in 71.2 % of the cases, the best result for a given layout is achieved on the Seven Categories corpus, which is characterized by a small number of documents m. In 28.8 % of the cases, the layout

Table 6: Layout algorithms resulting in the best result with respect to the accuracy metric α for each dataset.

Dataset	Layout	α
20 Newsgroups	(LSI,+,t-SNE,-), (LSI,+,t-SNE,+)	0.79
Emails	(LSI,+,t-SNE,-), (LSI,+,t-SNE,+)	0.76
GitHub Projects	(LSI,+,t-SNE,-),(LSI,+,t-SNE,+)	0.75
Reuters	(LDA,X,t-SNE,+)	0.66
Seven Categories	(LSI,-,t-SNE,-), (LSI,-,t-SNE,+)	0.78
	(LSI,+,t-SNE,-), (LSI,+,t-SNE,+)	
	(LSI,-,UMAP,-), (LSI,+,UMAP,-)	

Table 7: Layout algorithms resulting in the best result with respect to the perception metric β for each dataset.

Dataset	Layout	β
20 Newsgroups	(VSM,+,t-SNE,X), (LSI,+,t-SNE,+)	0.80
Emails	(VSM,+,t-SNE,X)	0.87
GitHub Projects	(VSM,-,t-SNE,X)	0.77
Reuters	(VSM,+,t-SNE,X)	0.69
Seven Categories	(VSM,+,t-SNE,X)	0.80

does not perform best on the Seven Categories corpus. In that case, 86.7 % use MDS as DR. Although the GitHub Projects corpus contains fewer documents, we suspect its high dimensionality n is why the best result for a layout is never achieved on the GitHub Projects corpus. With respect to β in 44.3 % of the rows, the Seven Categories corpus achieves the optimal result, followed by the Reuters (36.5 %) and 20 Newsgroup corpora (19.2 %), and the Emails (3.8 %). In no case, the optimal result is achieved on the GitHub Projects dataset. This order does not correspond to the increasing number of categories k. We assume Seven Categories performs best because of its small number of documents. We suspect that in these cases, the vocabulary has a more direct relationship to the topics, and thus more distinct clusters emerge when modeling the corpora using TMs.

4.3.3 Which DR technique performs best for a given TM?

Regarding α, for any of the given TMs, the best performing DR is either UMAP in 27.0 % of the cases or t-SNE in 81.0 % of the cases. A similar trend is obtained with respect to β. Here, t-SNE achieves the optimal result in 84.1 % of the cases and UMAP in 23.8 % of the cases. However, in the cases where UMAP is superior to t-SNE, their difference is negligible. With respect to α, in 47.5 % of the cases, the layout algorithms applying a linear combination according to Equation (2) perform equally to the direct application of the DR on the document representation. In 36.6 % of cases, using the linear combination improves the result, and in 15.9 % of cases, it performs less. Similar results are observed with respect to β. In 31.2 % of the cases, it improves the results. In 44.5 % of the cases, it matches the results and performs less in the remaining 24.3 % of the cases.

4.4 Influence of the Hyperparameters

In addition to the optimal values, it is particularly relevant how sensitive the DRs are to their hyperparameters. Figure 4 and Figure 5 show the five-number summaries for the quality metrics α and β merged over the datasets. The TMs are specified as a triple analogous to the description from above but with no indication of the DR. Figure 4 shows that for all layout algorithms that rely on MDS, α takes on values within a small range. Figure 5 shows the same pattern with respect to β. Even though a small range is desirable, MDS performs worse than the other DRs. For layouts derived from a SOM, α varies within a large range. The first and third quartiles are usually centered in the middle of the entire value range. Therefore, it is difficult to achieve good results when using a SOM. The range of values for β is more restricted in most cases. However, in cases of considerable variation, especially (LSI,-,-) and (LSI,-,+), the unfavorable location of the first and third quartiles can again be observed. For t-SNE, α also varies within a
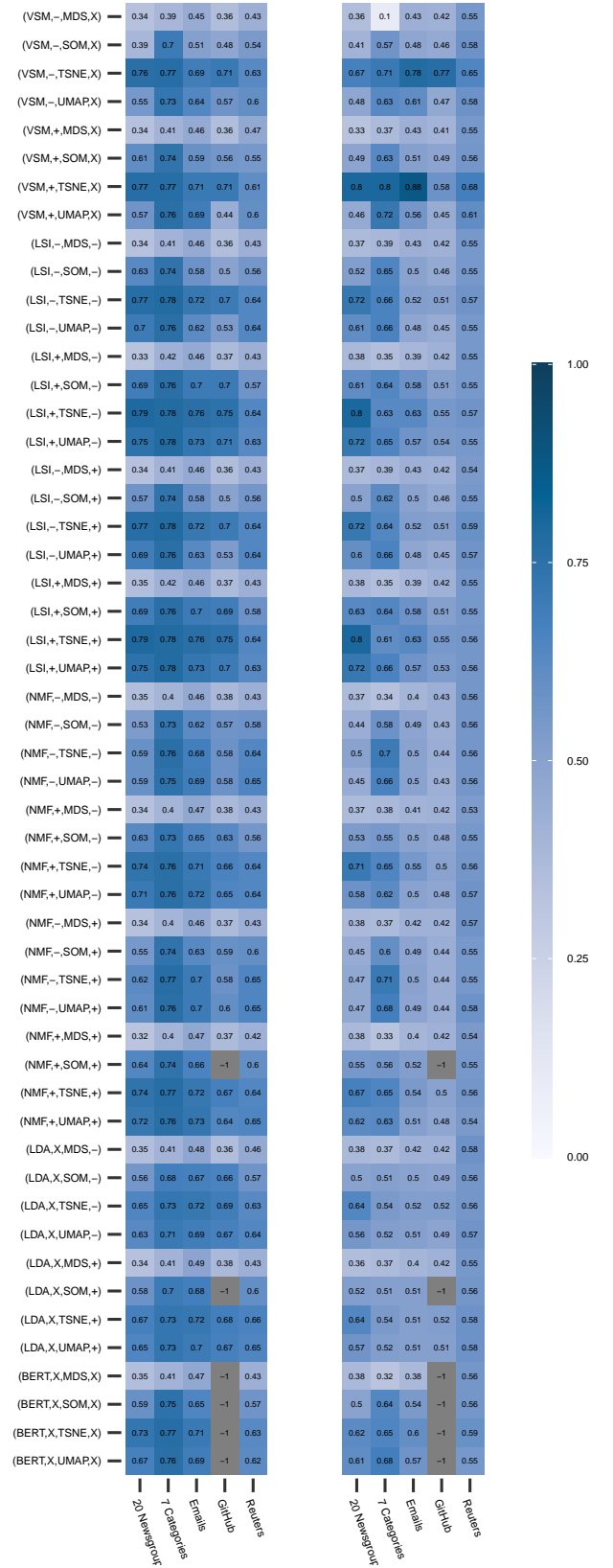
Fig. 3: Heatmap showing the best results for the layout algorithms on each dataset. The grey cells indicate combinations, where the TM could not be applied on the dataset. The second entry of quadruple specifying the layout indicates, whether the tf-idf weighting was applied (+), or not (-), or could not be applied (X). The fourth entry in the quadruple indicates whether Equation (2) was applied (+), or not (-), or could not be applied (X). (Left) Optimal values for the accuracy metric α, (Right) Optimal values for the perception metric β.

Fig. 4: Boxplots showing the summary statistic of the quality metric α.

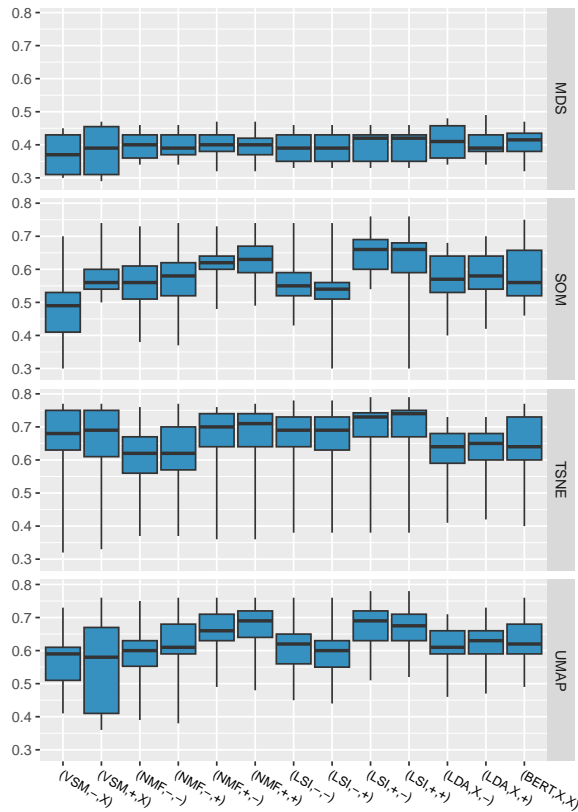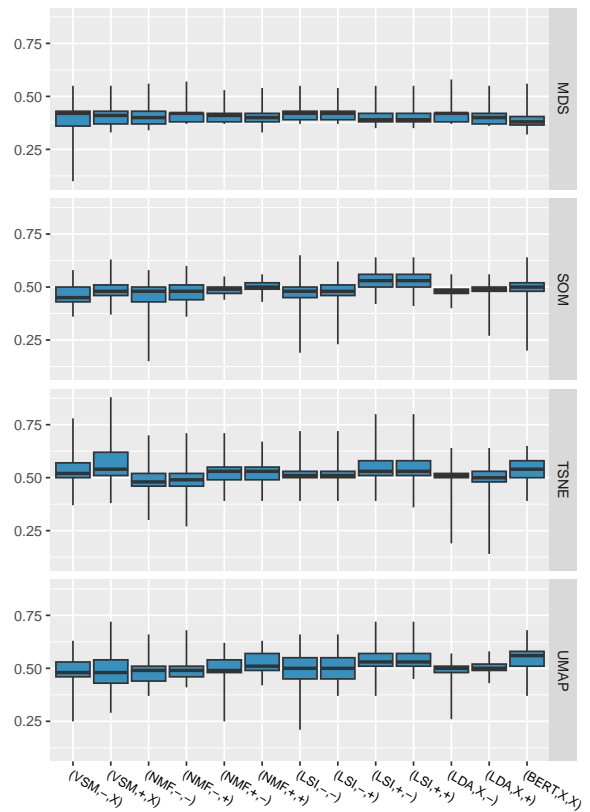Fig. 5: Boxplots showing the summary statistic of the quality metric β.

wide range. For all TMs, we observe that values above the first quartile are close to the optimum. From this, we conclude that with a high probability, any chosen hyperparameter setting achieves good results. Concerning β, the situation is different. Values below the third quartile are usually close to the minimum. Although the values for the first and third quartiles are similar to those of the SOMs, t-SNE shows more upside potential with respect to β. For UMAP, we observe similar patterns as for t-SNE with respect to α, albeit less pronounced and with apparent exceptions, e.g., (VSM,+,X). Regarding β, the location of the first and third quartiles resembles those of t-SNE. Concerning β, UMAP, and t-SNE perform similarly.

4.5 Performance of Default Hyperparameters

In typical application scenarios, the hyperparameters for a DR are left at their default values. Therefore, we analyzed the quality scores of the respective layouts compared to the other scores within our dataset. For each combination of TM, DR, and dataset, we determined the proportion of hyperparameter settings that yield better results for the metrics α and β than the default settings. The results, averaged over datasets, are summarized in Figure 6 for MDS, t-SNE, and UMAP. As the Sparse-SOM library has no default value for the number of neurons, we omitted this DR. MDS only requires the specification of the number of iterations. The implementation provided by Scikit-Learn specifies 300 as the default value. Across all TMs, we observe values close to the optimum. This indicates that MDS converges to a stable layout already after 300 iterations. t-SNE is known for being sensitive to its hyperparameters. Scikit-Learn specifies the default hyperparameters as `perplexity` = 30, `n_iter` = 1000, and `learning_rate` = max$(m/48,50)$, where m is the number of documents in our case. Only 13 % of the hyperparameter settings perform better than the default values with respect to α. With respect to β only 15 % achieve better results. UMAP-Learn specifies the default values to `n_neighbors` = 15 and `min_dist` = 0.1. In any case, maximal 13 % of the layouts perform better than the default settings with respect to α. With respect to β, maximal 16 % of the hyperparameter settings

achieve better results. Overall, the default values achieve good results.

5 DISCUSSION

From the results of our evaluation, we derive user guidelines for the effective combination of TMs and DRs. However, our benchmark, and the guidelines strictly derived from it, are subject to threats to validity.

5.1 Guidelines

One of the main goals for two-dimensional layouts for text corpora is the preservation of structures within the high-dimensional representation, and the separability of clusters in the low-dimensional representation. We captured both the accuracy metric α and the perception metric β, respectively. The tf-idf weighting is often applied on the DTM as an additional preprocessing step. Our first experiment indicates, on a formal statistical justification, that the tf-idf weighting tends to improve both the accuracy and the perception.

G1 When applying the LSI, NMF, or no TM, the DTM should be weighted according to the tf-idf scheme.

Analogously our binary tests revealed that applying Equation (2) improves the results.

G2 When applying the LSI, NMF, or LDA, the document positions should be aggregated according to Equation (2).

For each dataset, we trained exactly one version for each TM. Even without adjusting the hyperparameters of the TM, e.g., the number of topics, our second experiment showed that better results could be achieved when using a TM rather than solely relying on the VSM concerning α. Concerning β, a consecutive TM did not show improvements. However, as we did not investigate the full capabilities of the TM, we can not conclude whether TM can improve the results concerning β. Furthermore, the tf-idf weighting showed improvements concerning α, and usually the tf-idf weighting results in better interpretable topics. We deduce:

Accuracy α				Perception β		
MDS	t-SNE	UMAP	Topic Model	MDS	t-SNE	UMAP
0.00	0.08	0.10	(LSI,+,+)	0.00	0.12	0.05
0.10	0.07	0.10	(LSI,+,-)	0.00	0.12	0.16
0.00	0.06	0.01	(VSM,-,X)	0.00	0.07	0.11
0.00	0.10	0.12	(NMF,-,+)	0.00	0.08	0.00
0.00	0.07	0.10	(LSI,-,-)	0.00	0.07	0.01
0.00	0.11	0.11	(LDA,X,+)	0.00	0.12	0.04
0.00	0.13	0.11	(NMF,-,-)	0.00	0.12	0.00
0.00	0.07	0.10	(LSI,-,+)	0.00	0.07	0.00
0.00	0.08	0.12	(BERT,X,X)	0.00	0.15	0.13
0.00	0.00	0.00	(NMF,+,+)	0.00	0.09	0.02
0.00	0.00	0.00	(NMF,+,-)	0.00	0.07	0.02
0.00	0.04	0.02	(VSM,+,X)	0.00	0.08	0.12
0.00	0.12	0.12	(LDA,X,-)	0.00	0.03	0.16

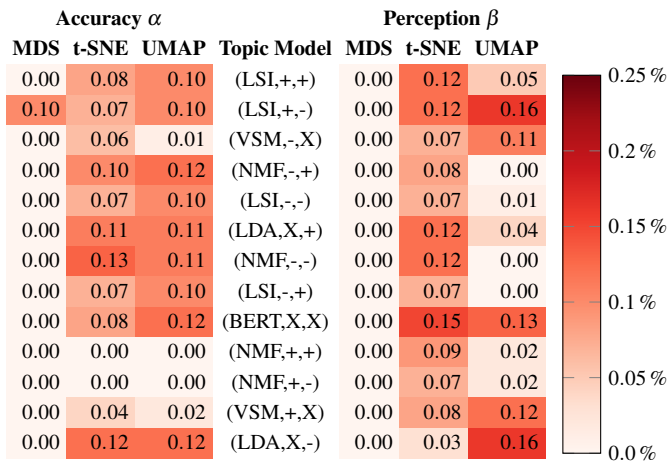Color scale: 0.25 %, 0.2 %, 0.15 %, 0.1 %, 0.05 %, 0.0 %

Fig. 6: Heatmap showing the percentage of resulting layouts that perform better than the default configuration for a given layout algorithm. (Left) Results according to the accuracy metric α, (Right) Results according to the perception metric β.

G3 A interpretable TM will probably improve the quality of a layout with respect to α.

For both α and β best results were achieved using t-SNE. Furthermore, our third experiment revealed that with respect to α most hyperparameter settings result in layouts near the optimum. The default values result in high-quality layouts.

G4 We recommend the use of t-SNE as DR with its default values.

In particular, the use of t-SNE is also recommended by Nonato and Aupetit for the analytics task "Explore Items in Base Layout" [55] and previous benchmarks [28].

5.2 Threats to Validity

Our design of the benchmark, its execution, the analysis of the results, and the derived guidelines are subject to internal and external threats to validity. We identify two kinds of internal threats to validity. The first concerns errors that may have occurred in evaluating the benchmark (Instrumentation). For example, our results may be subject to human errors in software development. For one, the risk is mitigated by using actively maintained open-source libraries for all algorithmic aspects that are often used in the ML community. Further, the source code was reviewed by at least one additional co-author. Last, we publish our entire implementation to allow for future improvements. The second internal threat concerns errors that may be caused by the adjustments made during the execution of the benchmark (Attrition). The layouts could be computed in ≈94.7 % of targeted cases. Those cases can be attributed to exceeding memory consumption. Even using the entire RAM of a node (≈400 GiB) has not prevented an out-of-memory event.

As the main external threat to validity, we identify the sampling bias. Our benchmark is subject to sampling bias with respect to our selection of datasets, TMs, DRs, hyperparameters, their value range, and quality metrics. In particular, it is unclear how our proposed guidelines are generalizable to other datasets according to the *no free lunch theorem* [1]. Furthermore, we trained one TM for each dataset with fixed hyperparameters. Even though we manually viewed the resulting topics and followed best practices, it is unclear how the choices of hyperparameters for the TM, e.g., the number of topics, affect the results. Furthermore, many of the layout algorithms sampled are not deterministic. It is unclear to what extent this affects the generated layouts. To mitigate this, we plan to evaluate multiple runs for a fixed hyperparameter configuration and evaluate average values with confidence intervals. Our choice of quality metrics was heavily influenced by previous benchmarking studies. However, we did not consider the *Normalized Stress* because, in many cases, it exceeds the range from 0 to 1, denoted as the value range in [28, 70]. To address the sampling bias, the benchmark is designed to be extensible with respect to $\mathcal{D}$, $\mathcal{L}$,

and $\mathcal{Q}$. Furthermore, we see great potential in quantifying the quality of our benchmark, e.g., by measuring the *Data Quality Index* proposed by Mishra et al. [52].

6 CONCLUSIONS & FUTURE WORK

Many visualizations for text corpora rely on a two-dimensional spatialization derived from combining a TM and a subsequent DR. Even though the choice of the TM, DR, and their respective hyperparameters significantly impacts the resulting layout, it is unknown how to obtain a two-dimensional layout reflecting both the structure within the corpus and the cluster separation between categories. We proposed a benchmark $\mathcal{B} = (\mathcal{D}, \mathcal{L}, \mathcal{Q})$ consisting of a set of text corpora $\mathcal{D}$, a set of layout algorithms $\mathcal{L}$ that are combinations of TMs and DRs, and a set of quality metrics $\mathcal{Q}$. We published our benchmark, which is also designed to be extensible for further experiments. By analyzing the correlation between the quality metrics, we defined an accuracy metric α, capturing preservation of high-dimensional structures in the layout, and a perception metric β, capturing separability between clusters. By extensive analysis, we discussed the results after executing our benchmark and derived guidelines for the effective use of TMs and DRs for generating two-dimensional layouts for text corpora. We recommend the use of LSI or the VSM, depending on whether the aggregated accuracy metric α or the aggregated perception metric β is to be optimized. The results can further be improved by applying the tf-idf weighting scheme. In our experiments, t-SNE has shown the overall best performance. In any case, the layout originating from the default hyperparameters is among the top 20 %. Unfortunately, none of the visualization approaches specified in Table 1 used our recommended layout algorithm. We hope practitioners and researchers consider our results and guidelines in their visualization design or extend the benchmark for a more evaluation.

For future work, we see different promising directions. We plan to expand our benchmark to address the major threats to validity. Also, we see potential in evaluating more variants of a TM, i.e., by iterating over its hyperparameter, or new variants of the BERT model, e.g., specialized for the case of source code. Furthermore, the temporal stability of a layout is particularly interesting for streaming text corpora, e.g., from social media. Our benchmark could be extended to incorporate time stability as proposed by Vernier et al. [70]. In addition to the quality metrics, the layouts are saved as well. We plan to analyze the relationship between DRs and their generated shapes in the two-dimensional representation. For this, we measure metrics related to the shape, e.g., the popular scagnostics [74, 75] or measures presented by Xia et al. [77]. This results in a dataset where each point describes the shape of a single scatter plot. By clustering this dataset and labeling the categories according to their visual characteristics, one can determine which DR will likely result in a specific shape, as recently presented by Machado et al. [49], and whether it is beneficial for a particular text visualization task. It would also be interesting to do a coding of resulting scatter plot patterns by means of an open coding study [56]. For a large corpus, computational costs are relevant. We consider evaluating the runtime of different layout algorithms. To improve computation times for layouts, neural networks can be trained on a set of precomputed layouts to approximate a given DR [27]. This, in particular, enables applications such as interactive exploration of the hyperparameter space of DR [6] or comparative analysis [32]. We plan to analyze to what extent our dataset is suitable for training a neural network that predicts layouts based on TMs for text corpora.

REFERENCES

[1] S. P. Adam, S.-A. N. Alexandropoulos, P. M. Pardalos, and M. N. Vrahatis. No free lunch theorem: A review. In *Approximation and Optimization: Algorithms, Complexity and Applications*, pp. 57–82. Springer, 2019. doi: 10.1007/978-3-030-12767-1_5 9

[2] C. C. Aggarwal and C. Zhai. A survey of text classification algorithms. In *Mining text data*, pp. 163–222. Springer, 2012. doi: 10.1007/978-1-4614 -3223-4_6 2

[3] C. C. Aggarwal and C. Zhai. A survey of text clustering algorithms. In *Mining Text Data*, pp. 77–128. Springer, 2012. doi: 10.1007/978-1-4614 -3223-4_4 2, 4

[4] G. Albuquerque, M. Eisemann, and M. Magnor. Perception-based visual quality measures. In *Proc. Conference on Visual Analytics Science and Technology*, VAST '11, pp. 13–20. IEEE, 2011. doi: 10.1109/VAST.2011. 6102437 3

[5] E. Alexander and M. Gleicher. Task-driven comparison of topic models. *IEEE TVCG*, 22(1):320–329, 2016. doi: 10.1109/TVCG.2015.2467618 2

[6] G. Appleby, M. Espadoto, R. Chen, S. Goree, A. C. Telea, E. W. Anderson, and R. Chang. HyperNP: Interactive visual exploration of multidimensional projection hyperparameters. *EG Computer Graphics Forum*, 41(3):169–181, 2022. doi: 10.1111/cgf.14531 9

[7] D. Atzberger, T. Cech, M. de la Haye, M. Söchting, W. Scheibel, D. Limberger, and J. Döllner. Software Forest: A visualization of semantic similarities in source code using a tree metaphor. In *Proc. 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications – Volume 3*, IVAPP '21, pp. 112–122. INSTICC, SciTePress, 2021. doi: 10.5220/0010267601120122 2, 4

[8] D. Atzberger, T. Cech, A. Jobst, W. Scheibel, D. Limberger, M. Trapp, and J. Döllner. Visualization of knowledge distribution across development teams using 2.5D semantic software maps. In *Proc. 17th International Conference on Information Visualization Theory and Applications – Volume 3*, IVAPP '22, pp. 210–217. INSTICC, SciTePress, 2022. doi: 10. 5220/0010991100003124 2

[9] D. Atzberger, T. Cech, W. Scheibel, D. Limberger, J. Döllner, and M. Trapp. A benchmark for the use of topic models for text visualization tasks. In *Proc. 15th International Symposium on Visual Information Communication and Interaction*, VINCI '22. ACM, 2022. Poster Presentation. doi: 10.1145/3554944.3554961 3

[10] D. Atzberger, N. Scordialo, T. Cech, W. Scheibel, M. Trapp, and J. Döllner. CodeCV: Mining expertise of GitHub users from coding activities. In *Proc. 22nd International Working Conference on Source Code Analysis and Manipulation*, SCAM '22, pp. 143–147. IEEE, 2022. doi: 10.1109/ SCAM55253.2022.00021 3

[11] M. Behrisch, M. Blumenschein, N. W. Kim, L. Shao, M. El-Assady, J. Fuchs, D. Seebacher, A. Diehl, U. Brandes, H. Pfister, T. Schreck, D. Weiskopf, and D. A. Keim. Quality metrics for information visualization. *EG Computer Graphics Forum*, 37(3):625–662, 2018. doi: 10. 1111/cgf.13446 2

[12] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003. doi: 10.5555/944919. 944937 4

[13] P. Caillou, J. Renault, J.-D. Fekete, A.-C. Letournel, and M. Sebag. Cartolabe: A web-based scalable visualization of large document collections. *IEEE CG&A*, 41(2):76–88, 2021. doi: 10.1109/MCG.2020.3033401 2

[14] T. Caliński and J. Harabasz. A dendrite method for cluster analysis. *Taylor & Francis Communications in Statistics*, 3(1):1–27, 1974. doi: 10.1080/ 03610927408827101 5

[15] T.-H. Chen, S. W. Thomas, and A. E. Hassan. A survey on the use of topic models when mining software repositories. *Springer Empirical Software Engineering*, 21(5):1843–1919, 2016. doi: 10.1007/s10664-015-9402-8 3

[16] Y. Chen, L. Wang, M. Dong, and J. Hua. Exemplar-based visualization of large document corpus. *IEEE TVCG*, 15(6):1161–1168, 2009. doi: 10. 1109/TVCG.2009.140 3

[17] J. Choo, C. Lee, C. K. Reddy, and H. Park. UTOPIAN: User-driven topic modeling based on interactive nonnegative matrix factorization. *IEEE TVCG*, 19(12):1992–2001, 2013. doi: 10.1109/TVCG.2013.212 2

[18] M. A. A. Cox and T. F. Cox. Multidimensional scaling. In *Handbook of Data Visualization*, pp. 315–347. Springer, 2008. doi: 10.1007/978-3-540 -33037-0_14 4

[19] S. P. Crain, K. Zhou, S.-H. Yang, and H. Zha. Dimensionality reduction and topic modeling: From latent semantic indexing to latent Dirichlet allocation and beyond. In *Mining Text Data*, pp. 129–161. Springer, 2012.

[20] J. P. Cunningham and Z. Ghahramani. Linear dimensionality reduction: Survey, insights, and generalizations. *Journal of Machine Learning Research*, 16(1):2859–2900, 2015. 2

[21] T. Dang, H. N. Nguyen, V. Pham, J. Johansson, F. Sadlo, and G. Marai. WordStream: Interactive visualization for topic evolution. In *Proc. European Conference on Visualization*, EuroVis '19, pp. 103–107. EG, 2019. Poster Presentation. doi: 10.2312/evs.20191178 2

[22] D. L. Davies and D. W. Bouldin. A cluster separation measure. *IEEE TPAMI*, PAMI-1(2):224–227, 1979. doi: 10.1109/TPAMI.1979.4766909 5

[23] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990. doi: 10.1002/(SICI) 1097-4571(199009)41:6%3C391::AID-ASI1%3E3.0.CO;2-9 4

[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. North-American Chapter of the Association for Computational Linguistics*, NAACL-HLT '19, pp. 4171–4186. ACL, 2019. 4

[25] W. Dou, L. Yu, X. Wang, Z. Ma, and W. Ribarsky. HierarchicalTopics: Visually exploring large text collections using topic hierarchies. *IEEE TVCG*, 19(12):2002–2011, 2013. doi: 10.1109/TVCG.2013.162 2

[26] D. Engel, L. Hüttenberger, and B. Hamann. A survey of dimension reduction methods for high-dimensional data analysis and visualization. In *Proc. Workshop on Visualization of Large and Unstructured Data Sets: Applications in Geospatial Planning, Modeling and Engineering*, pp. 135–149. DROPS, 2012. doi: 10.4230/OASIcs.VLUDS.2011.135 2

[27] M. Espadoto, N. S. T. Hirata, and A. C. Telea. Deep learning multidimensional projections. *SAGE Information Visualization*, 19(3):247–269, 2020. doi: 10.1177/1473871620909485 9

[28] M. Espadoto, R. M. Martins, A. Kerren, N. S. T. Hirata, and A. C. Telea. Toward a quantitative survey of dimension reduction techniques. *IEEE TVCG*, 27(3):2153–2173, 2021. doi: 10.1109/TVCG.2019.2944182 3, 4, 5, 6, 9

[29] M. Espadoto, E. F. Vernier, and A. C. Telea. Selecting and sharing multidimensional projection algorithms: a practical view. In *Proc. Workshop on the Gap between Visualization Research and Visualization Software*, VisGap '20, pp. 9–16. EG, 2020. doi: 10.2312/visgap20201105 3

[30] I. K. Fodor. A survey of dimension reduction techniques. Technical Report UCRL-ID-148494, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 2002. 2

[31] D. Fried and S. G. Kobourov. Maps of computer science. In *Proc. Pacific Visualization Symposium*, PacificVis '14, pp. 113–120. IEEE, 2014. doi: 10.1109/PacificVis.2014.47 2

[32] T. Fujiwara, X. Wei, J. Zhao, and K.-L. Ma. Interactive dimensionality reduction for comparative analysis. *IEEE TVCG*, 28(1):758–768, 2022. doi: 10.1109/TVCG.2021.3114807 9

[33] E. R. Gansner, Y. Hu, and S. C. North. Interactive visualization of streaming text data with dynamic maps. *Brown Journal of Graph Algorithms and Applications*, 17(4):515–540, 2013. doi: 10.7155/jgaa.00302 2

[34] F. J. García Fernández, M. Verleysen, J. A. Lee, I. Díaz Blanco, et al. Stability comparison of dimensionality reduction techniques attending to data and parameter variations. In *Proc. Workshop on Visual Analytics using Multidimensional Projections*, VAMP '13. EG, 2013. doi: 10.2312/PE.VAMP .VAMP2013.005-009 3

[35] A. Gisbrecht and B. Hammer. Data visualization by nonlinear dimensionality reduction. *Wiley Data Mining and Knowledge Discovery*, 5(2):51–73, 2015. doi: 10.1002/widm.1147 3

[36] M. Hográfer, M. Heitzler, and H.-J. Schulz. The state of the art in map-like visualization. *EG Computer Graphics Forum*, 39(3):647–674, 2020. doi: 10.1111/cgf.14031 1

[37] S. Ingram and T. Munzner. Dimensionality reduction for documents with nearest neighbor queries. *Neurocomputing*, 150:557–569, 2015. doi: 10. 1016/j.neucom.2014.07.073 3

[38] P. Joia, D. Coimbra, J. A. Cuminato, F. V. Paulovich, and L. G. Nonato. Local affine multidimensional projection. *IEEE TVCG*, 17(12):2563–2571, 2011. doi: 10.1109/TVCG.2011.220 5

[39] I. Jolliffe. Principal component analysis. In *Encyclopedia of Statistics in Behavioral Science*. John Wiley & Sons, Ltd, 2005. doi: 10.1002/ 0470013192.bsa501 4

[40] M. Kim, K. Kang, D. Park, J. Choo, and N. Elmqvist. TopicLens: Efficient multi-level visual topic exploration of large-scale document collections. *IEEE TVCG*, 23(1):151–160, 2017. doi: 10.1109/TVCG.2016.2598445 2

[41] T. Kohonen. Exploration of very large databases by self-organizing maps.

doi: 10.1007/978-1-4614-3223-4_5 2, 4

In *Proc. International Conference on Neural Networks*, ICNN '97, pp. 1–6. IEEE, 1997. doi: 10.1109/ICNN.1997.611622 4

[42] K. Kucher and A. Kerren. Text visualization revisited: The state of the field in 2019. In *Proc. European Conference on Visualization*, EuroVis '19, pp. 29–31. EG, 2019. Poster Presentation. doi: 10.2312/eurp.20191138 1

[43] K. Kucher, R. M. Martins, and A. Kerren. Analysis of VINCI 2009–2017 proceedings. In *Proc. 11th International Symposium on Visual Information Communication and Interaction*, VINCI '18, pp. 97–101. ACM, 2018. doi: 10.1145/3231622.3231641 2

[44] A. Kuhn, D. Erni, P. Loretan, and O. Nierstrasz. Software cartography: Thematic software visualization with consistent layout. *Journal of Software Maintenance and Evolution: Research and Practice*, 22(3):191–210, 2010. doi: 10.1002/smr.414 2

[45] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999. doi: 10.1038/44565 4

[46] D. J. Lehmann and H. Theisel. Optimal sets of projections of high-dimensional data. *IEEE TVCG*, 22(1):609–618, 2016. doi: 10.1109/TVCG.2015.2467132 2

[47] E. Linstead, S. Bajracharya, T. Ngo, P. Rigor, C. Lopes, and P. Baldi. Sourcerer: Mining and searching internet-scale software repositories. *Springer Data Mining and Knowledge Discovery*, 18(2):300–336, 2009. doi: 10.1007/s10618-008-0118-x 2

[48] Z. C. Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *ACM Queue*, 16(3):31–57, 2018. doi: 10.1145/3236386.3241340 2

[49] A. Machado, A. Telea, and M. Behrisch. ShaRP: Shape-regularized multidimensional projections. In *EuroVis Workshop on Visual Analytics*, EuroVA '23. The Eurographics Association, 2023. doi: 10.2312/eurova.20231088 9

[50] L. McInnes, J. Healy, and J. Melville. UMAP: Uniform manifold approximation and projection for dimension reduction. *arXiv CoRR*, stat.ML(1802.03426):1–63, 2020. pre-print. doi: 10.48550/arXiv.1802.03426 4

[51] J. Melka and J.-J. Mariage. Adapting self-organizing map algorithm to sparse data. In *Computational Intelligence*, pp. 139–161. Springer, 2019. doi: 10.1007/978-3-030-16469-0_8 5

[52] S. Mishra, A. Arunkumar, B. Sachdeva, C. Bryan, and C. Baral. DQI: A guide to benchmark evaluation. *arXiv CoRR cs.CL*, arXiv:2008.03964, 2020. doi: 10.48550/arXiv.2008.03964 9

[53] C. Morariu, A. Bibal, R. Cutura, B. Frénay, and M. Sedlmair. Predicting user preferences of dimensionality reduction embedding quality. *IEEE TVCG*, 29(1):745–755, 2023. doi: 10.1109/TVCG.2022.3209449 2, 3, 5

[54] A. Nenkova and K. McKeown. A survey of text summarization techniques. In *Mining Text Data*, pp. 43–76. Springer, 2012. doi: 10.1007/978-1-4614-3223-4_3 2

[55] L. G. Nonato and M. Aupetit. Multidimensional projection for visual analytics: Linking techniques with distortions, tasks, and layout enrichment. *IEEE TVCG*, 25(8):2650–2673, 2019. doi: 10.1109/TVCG.2018.2846735 1, 2, 9

[56] A. V. Pandey, J. Krause, C. Felix, J. Boy, and E. Bertini. Towards understanding human similarity perception in the analysis of large sets of scatter plots. In *Proc. CHI Conference on Human Factors in Computing Systems*, CHI '16, pp. 3659–3669. ACM, 2016. doi: 10.1145/2858036.2858155 9

[57] F. Paulovich and R. Minghim. Text map explorer: a tool to create and explore document maps. In *Proc. 10th International Conference on Information Visualisation*, IV '06, pp. 245–251. IEEE, 2006. doi: 10.1109/IV.2006.104 4

[58] J. Peter, S. Szigeti, A. Jofre, and S. Diamond. Topicks: Visualizing complex topic models for user comprehension. In *Proc. Conference on Visual Analytics Science and Technology*, VAST '15, pp. 207–208. IEEE, 2015. doi: 10.1109/VAST.2015.7347681 2

[59] P. Riehmann, D. Kiesel, M. Kohlhaas, and B. Froehlich. Visualizing a thinker's life. *IEEE TVCG*, 25(4):1803–1816, 2019. doi: 10.1109/TVCG.2018.2824822 2

[60] M. Röder, A. Both, and A. Hinneburg. Exploring the space of topic coherence measures. In *Proc. 8th International Conference on Web Search and Data Mining*, WSDM '15, pp. 399–408. ACM, 2015. doi: 10.1145/2684822.2685324 2

[61] P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Elsevier Journal of Computational and Applied Mathematics*, 20:53–65, 1987. doi: 10.1016/0377-0427(87)90125-7

5

[62] M. Sedlmair and M. Aupetit. Data-driven evaluation of visual quality measures. *EG Computer Graphics Forum*, 34(3):201–210, 2015. doi: 10.1111/cgf.12632 3, 5

[63] M. Sedlmair, T. Munzner, and M. Tory. Empirical guidance on scatterplot and dimension reduction technique choices. *IEEE TVCG*, 19(12):2634–2643, 2013. doi: 10.1109/TVCG.2013.153 3

[64] C. Sievert and K. Shirley. LDAvis: A method for visualizing and interpreting topics. In *Proc. Workshop on Interactive Language Learning, Visualization, and Interfaces*, pp. 63–70. ACL, 2014. doi: 10.3115/v1/W14-3110 2

[65] M. Sips, B. Neubert, J. P. Lewis, and P. Hanrahan. Selecting good views of high-dimensional data using class consistency. *EG Computer Graphics Forum*, 28(3):831–838, 2009. doi: 10.1111/j.1467-8659.2009.01467.x 5

[66] A. Skupin. The world of geography: Visualizing a knowledge domain with cartographic means. *Proceedings of the National Academy of Sciences*, 101(suppl_1):5274–5278, 2004. doi: 10.1073/pnas.0307654100 2

[67] L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11):2579–2605, 2008. 4

[68] L. van der Maaten, E. Postma, and J. van den Herik. Dimensionality reduction: a comparative review. Technical Report 009–005, Tilburg University, Tilburg Centre for Creative Computing, The Netherlands, 2009. 2

[69] J. Venna and S. Kaski. Visualizing gene interaction graphs with local multidimensional scaling. In *Proc. European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, ESANN '06, pp. 557–562. ESANN, 2006. 4

[70] E. F. Vernier, J. L. D. Comba, and A. C. Telea. Guided stable dynamic projections. In *EG Computer Graphics Forum*, vol. 40, pp. 87–98, 2021. doi: 10.1111/cgf.14291 3, 4, 5, 9

[71] E. F. Vernier, R. Garcia, I. d. Silva, J. L. D. Comba, and A. C. Telea. Quantitative evaluation of time-dependent multidimensional projection techniques. In *EG Computer Graphics Forum*, vol. 39, pp. 241–252, 2020. doi: 10.1111/cgf.13977 3, 4, 5

[72] H. M. Wallach, D. Mimno, and A. K. McCallum. Rethinking LDA: Why priors matter. In *Proc. 22nd International Conference on Neural Information Processing Systems*, NIPS '09, pp. 1973–1981, 2009. 5

[73] Y. Wang, K. Feng, X. Chu, J. Zhang, C.-W. Fu, M. Sedlmair, X. Yu, and B. Chen. A perception-driven approach to supervised dimensionality reduction for visualization. *IEEE TVCG*, 24(5):1828–1840, 2018. doi: 10.1109/TVCG.2017.2701829 3

[74] L. Wilkinson, A. Anand, and R. Grossman. Graph-theoretic scagnostics. In *Proc. Symposium on Information Visualization*, InfoVis '05, pp. 157–164. IEEE, 2005. doi: 10.1109/INFVIS.2005.1532142 9

[75] L. Wilkinson, A. Anand, and R. Grossman. High-dimensional visual analytics: Interactive exploration guided by pairwise views of point distributions. *IEEE TVCG*, 12(6):1363–1372, 2006. doi: 10.1109/TVCG.2006.94 9

[76] J. Xia, L. Huang, W. Lin, X. Zhao, J. Wu, Y. Chen, Y. Zhao, and W. Chen. Interactive visual cluster analysis by contrastive dimensionality reduction. *IEEE TVCG*, 29(1):734–744, 2023. doi: 10.1109/TVCG.2022.3209423 3

[77] J. Xia, W. Lin, G. Jiang, Y. Wang, W. Chen, and T. Schreck. Visual clustering factors in scatterplots. *IEEE CG&A*, 41(5):79–89, 2021. doi: 10.1109/MCG.2021.3098804 3, 9

[78] J. Xia, Y. Zhang, J. Song, Y. Chen, Y. Wang, and S. Liu. Revisiting dimensionality reduction techniques for visual cluster analysis: An empirical study. *IEEE TVCG*, 28(1):529–539, 2022. doi: 10.1109/TVCG.2021.3114694 3

[79] P. Xu, Y. Wu, E. Wei, T.-Q. Peng, S. Liu, J. J. Zhu, and H. Qu. Visual analysis of topic competition on social media. *IEEE TVCG*, 19(12):2012–2021, 2013. doi: 10.1109/TVCG.2013.221 2

[80] Y. Yan, Y. Tao, S. Jin, J. Xu, and H. Lin. An interactive visual analytics system for incremental classification based on semi-supervised topic modeling. In *Proc. Pacific Visualization Symposium*, PacificVis '19, pp. 148–157. IEEE, 2019. doi: 10.1109/PacificVis.2019.00025 2

[81] A. B. Yoo, M. A. Jette, and M. Grondona. SLURM: Simple linux utility for resource management. In *Proc. Workshop on Job Scheduling Strategies for Parallel Processing*, JSSPP '03, pp. 44–60. Springer, 2003. doi: 10.1007/10968987_3 5

Large-Scale Evaluation of Topic Models and Dimensionality Reduction Methods for 2D Text Spatialization
Supplemental Material

Daniel Atzberger (iD), Tim Cech (iD), Matthias Trapp (iD), Rico Richter (iD),
Willy Scheibel (iD), Jürgen Döllner (iD), and Tobias Schreck (iD)

A TOPICS

Table 1 – Table 29 show the top ten words for selected topics of all considered Topic Models.

B EXAMPLE LAYOUTS

Figure 1 shows the layouts generated from (LSI,+,TSNE,+) and (VSM,+,TSNE,X) for all datasets respectively using the default parameters of t-SNE. The color represents the class label.

Table 1: Top ten words for ten selected topics for the LSI model for the 20 Newsgroups dataset.

0	1	2	3	4	5	6	7	8	9
would	window	key	key	drive	god	card	drive	car	space
use	file	game	god	window	gun	driver	gun	window	file
get	god	chip	game	file	car	drive	window	bike	henry
people	card	team	team	ide	card	file	file	god	gun
one	drive	clipper	chip	car	key	gun	ide	key	sale
say	driver	encryption	clipper	controller	game	video	disk	driver	zoo
window	people	player	encryption	card	drive	disk	sale	gun	window
know	disk	win	player	disk	ide	window	mail	space	car
go	video	play	window	program	bike	mouse	monitor	mouse	god
think	color	escrow	escrow	win	chip	diamond	god	ride	mail

Table 2: Top ten words for the eight topics for the LSI model for the Emails dataset.

0	1	2	3	4	5	6	7
key	key	bit	key	stratus	phone	access	gun
chip	chip	government	stratus	transfer	tap	net	genocide
encryption	clipper	encryption	bit	key	public	pat	turk
government	encryption	space	net	rocket	clipper	key	chip
clipper	escrow	key	access	gun	line	stratus	clipper
use	bit	law	encryption	message	encryption	encryption	soviet
would	gun	stratus	gun	bit	serial	chip	stratus
people	algorithm	clipper	clipper	distribution	warrant	escrow	firearm
phone	phone	gun	technology	carl	file	reference	bony
system	serial	privacy	host	encrypt	security	space	law

Table 3: Top ten words for ten selected topics for the LSI model for the GitHub projects dataset.

0	1	2	3	4	5	6	7	8	9
texture	texture	tensor	webpack	price	mut	mut	wallet	webpack	react
prototype	vec	torch	react	market	webpack	webpack	blockchain	react	webpack
vec	shader	train	chart	candle	pub	pub	transaction	err	err
webpack	vertex	np	vue	trade	err	chart	mut	syscall	jest
vertex	mesh	webpack	pub	exchange	wallet	err	boost	sqlite	sqlite
mesh	err	shape	mut	buy	react	plot	err	plot	vue
tensor	gl	tf	vec	binance	crate	crate	chart	jest	syscall
model	geometry	react	texture	mut	func	axis	std	std	eslint
shader	material	err	wallet	pub	chart	fn	pub	prop	sql
std	tensor	model	err	symbol	vec	react	block	vue	chart

Table 4: Top ten words for ten selected topics for the LSI model for the Reuters dataset.

0	1	2	3	4	5	6	7	8	9
loss	loss	net	div	billion	profit	share	bank	split	split
net	net	div	billion	share	loss	billion	rate	dividend	gain
profit	billion	record	prior	profit	billion	split	oil	quarter	quarter
nine	profit	loss	record	stock	tax	stock	sale	rise	oil
year	div	prior	pay	div	bank	rate	prime	offer	stock
sale	record	pay	net	offer	loan	bank	split	year	exclude
note	prior	dividend	bank	net	nil	dividend	money	corp	crude
billion	nine	set	quarterly	rise	net	common	price	group	sale
include	pay	march	dividend	say	share	oil	year	stake	include
gain	sale	quarterly	say	common	three	rise	billion	div	tax

Table 5: Top ten words for ten selected topics for the LSI model for the Seven Categories dataset.

0	1	2	3	4	5	6	7	8	9
instruction	share	share	war	share	cache	cell	debenture	charge	fraction
register	instruction	debenture	charge	debenture	instruction	cache	cash	motion	decimal
memory	register	allotment	soviet	cash	miss	debenture	sale	wave	number
bit	debenture	instruction	magnetic	purchase	block	plant	account	magnetic	bit
address	memory	cash	united	account	register	cash	cell	particle	lesson
cache	cash	charge	party	sale	memory	charge	purchase	field	exponent
branch	bit	application	government	allotment	page	water	plant	electric	denominator
charge	address	money	president	balance	branch	organism	redemption	current	instruction
cycle	allotment	capital	field	redemption	charge	miss	error	velocity	multiply
clock	charge	force	electric	book	hierarchy	magnetic	issue	cell	review

Table 6: Top ten words for ten selected topics for the LSI model with tfidf weighting for the 20 Newsgroups dataset.

0	1	2	3	4	5	6	7	8	9
would	window	key	key	drive	god	card	drive	car	space
use	file	game	god	window	gun	driver	gun	window	file
get	god	chip	game	file	car	drive	window	bike	henry
people	card	team	team	ide	card	file	file	god	gun
one	drive	clipper	chip	car	key	gun	ide	key	sale
say	driver	encryption	clipper	controller	game	video	disk	driver	zoo
window	people	player	encryption	card	drive	disk	sale	gun	window
know	disk	win	player	disk	ide	window	mail	space	car
go	video	play	window	program	bike	mouse	monitor	mouse	god
think	color	escrow	escrow	win	chip	diamond	god	ride	mail

Table 7: Top ten words for the eight topics for the LSI model with tfidf weighting for the Emails dataset.

0	1	2	3	4	5	6	7
key	key	bit	key	stratus	phone	access	gun
chip	chip	government	stratus	transfer	tap	net	genocide
encryption	clipper	encryption	bit	key	public	pat	turk
government	encryption	space	net	rocket	clipper	key	chip
clipper	escrow	key	access	gun	line	stratus	clipper
use	bit	law	encryption	message	encryption	encryption	soviet
would	gun	stratus	gun	bit	serial	chip	stratus
people	algorithm	clipper	clipper	distribution	warrant	escrow	firearm
phone	phone	gun	technology	carl	file	reference	bony
system	serial	privacy	host	encrypt	security	space	law

Table 8: Top ten words for ten selected topics for the LSI model with tfidf weighting for the GitHub projects dataset.

0	1	2	3	4	5	6	7	8	9
texture	texture	tensor	webpack	price	mut	mut	wallet	webpack	react
prototype	vec	torch	react	market	webpack	webpack	blockchain	react	webpack
vec	shader	train	chart	candle	pub	pub	transaction	err	err
webpack	vertex	np	vue	trade	err	chart	mut	syscall	jest
vertex	mesh	webpack	pub	exchange	wallet	err	boost	sqlite	sqlite
mesh	err	shape	mut	buy	react	plot	err	plot	vue
tensor	gl	tf	vec	binance	crate	crate	chart	jest	syscall
model	geometry	react	texture	mut	func	axis	std	std	eslint
shader	material	err	wallet	pub	chart	fn	pub	prop	sql
std	tensor	model	err	symbol	vec	react	block	vue	chart

Table 9: Top ten words for ten selected topics for the LSI model with tfidf weighting for the Reuters dataset.

0	1	2	3	4	5	6	7	8	9
loss	loss	net	div	billion	profit	share	bank	split	split
net	net	div	billion	share	loss	billion	rate	dividend	gain
profit	billion	record	prior	profit	billion	split	oil	quarter	quarter
nine	profit	loss	record	stock	tax	stock	sale	rise	oil
year	div	prior	pay	div	bank	rate	prime	offer	stock
sale	record	pay	net	offer	loan	bank	split	year	exclude
note	prior	dividend	bank	net	nil	dividend	money	corp	crude
billion	nine	set	quarterly	rise	net	common	price	group	sale
include	pay	march	dividend	say	share	oil	year	stake	include
gain	sale	quarterly	say	common	three	rise	billion	div	tax

Table 10: Top ten words for ten selected topics for the LSI model with tfidf weighting for the Seven Categories dataset.

0	1	2	3	4	5	6	7	8	9
instruction	share	share	war	share	cache	cell	debenture	charge	fraction
register	instruction	debenture	charge	debenture	instruction	cache	cash	motion	decimal
memory	register	allotment	soviet	cash	miss	debenture	sale	wave	number
bit	debenture	instruction	magnetic	purchase	block	plant	account	magnetic	bit
address	memory	cash	united	account	register	cash	cell	particle	lesson
cache	cash	charge	party	sale	memory	charge	purchase	field	exponent
branch	bit	application	government	allotment	page	water	plant	electric	denominator
charge	address	money	president	balance	branch	organism	redemption	current	instruction
cycle	allotment	capital	field	redemption	charge	miss	error	velocity	multiply
clock	charge	force	electric	book	hierarchy	magnetic	issue	cell	review

Table 11: Top ten words for ten selected topics for the LDA model for the 20 Newsgroups dataset.

0	1	2	3	4	5	6	7	8	9
gun	say	would	subject	use	use	drive	go	game	god
space	people	write	line	line	line	organization	say	team	church
year	one	think	organization	window	subject	subject	one	year	henry
weapon	know	subject	system	subject	organization	line	would	player	believe
firearm	would	article	post	organization	get	get	get	good	subject
orbit	time	like	book	file	need	university	people	subject	say
control	see	organization	use	card	one	post	know	organization	line
rate	think	line	mail	problem	also	would	well	line	organization
use	come	make	university	bit	look	write	make	write	one
new	write	say	computer	get	write	article	think	get	write

Table 12: Top ten words for the eight topics for the LDA model for the Emails dataset.

0	1	2	3	4	5	6	7
write	write	people	president	would	homosexual	would	would
gay	article	write	government	one	write	use	write
say	post	right	say	make	one	one	one
think	get	say	new	go	use	government	year
one	say	gun	people	write	article	law	article
go	would	article	go	people	bit	make	make
use	go	would	use	know	health	key	soviet
see	host	think	human	think	get	write	apartment
article	one	one	would	article	know	people	million
people	fire	go	write	say	would	say	

Table 13: Top ten words for ten selected topics for the LDA model for the GitHub projects dataset.

0	1	2	3	4	5	6	7	8	9
get	get	view	type	type	type	builder	wx	resource	sph
webpack	type	value	boost	webpack	get	info	model	name	gl
value	string	pyx	get	get	value	get	i	get	get
type	value	get	err	node	model	protobuf	d	value	value
node	size	object	value	order	size	com	string	xamarin	type
set	node	set	name	i	data	outer	get	set	set
name	set	type	string	module	name	grasscutter	std	type	index
key	name	insert	i	io	gl	net	mm	ly	size
ag	std	d	d	value	input	google	item	droid	name
i	d	name	set	d	set	emu	event	wd	string

Table 14: Top ten words for ten selected topics for the LDA model for the Reuters dataset.

0	1	2	3	4	5	6	7	8	9
oil	stock	billion	say	say	rate	say	say	price	price
energy	say	profit	rate	company	debenture	talk	would	raise	official
say	cocoa	net	currency	year	standard	fire	official	oil	japan
pipeline	buffer	year	market	quarter	effective	ship	oil	crude	output
barrel	delegate	loss	dollar	earning	say	preliminary	meeting	barrel	chip
crude	buy	sale	bank	share	split	company	make	soviet	say
day	manager	company	west	first	discount	south	plan	increase	reed
spot	rule	note	exchange	expect	plan	cause	take	say	cut
ford	consumer	nine	monetary	report	treasury	bankruptcy	import	post	market
york	purchase	operating	german	result	balance	report	country	west	fall

Table 15: Top ten words for ten selected topics for the LDA model for the Seven Categories dataset.

0	1	2	3	4	5	6	7	8	9
number	page	mass	city	cache	address	bit	execution	clock	instruction
point	hazard	energy	state	memory	force	reg	program	instruction	memory
magnetic	memory	nucleus	war	miss	block	register	branch	use	time
field	performance	atom	new	datum	particle	one	file	cycle	pipeline
two	use	two	also	use	body	use	designer	one	bit
use	time	error	two	write	system	time	address	figure	computer
force	program	electron	country	time	register	figure	system	memory	motion
time	code	decay	form	address	mass	two	also	state	use
vector	system	neutron	wave	register	data	also	force	exception	load
direction	force	page	south	water	procedure	call	state	two	two

Table 16: Top ten words for ten selected topics for the NMF model for the 20 Newsgroups dataset.

0	1	2	3	4	5	6	7	8	9
use	ax	god	use	file	would	health	year	ax	file
argument	di	write	say	gun	one	would	line	um	mail
one	ey	subject	one	firearm	people	center	line	mu	mail
drive	biz	people	time	control	say	medical	get	ey	use
make	ex	organization	people	bill	know	use	output	mu	send
fallacy	mu	article	year	handgun	make	think	cancer	ah	graphic
would	ne	father	health	law	write	cancer	write	mi	gun
post	mi	say	get	united	write	mu	use	ex	format
see	om	son	work	weapon	well	tobacco	program	pu	system
problem	ah	make	go	crime	use	new	file	biz	ray

Table 21: Top ten words for ten selected topics for the NMF model with tfidf weighting for the 20 Newsgroups dataset.

0	1	2	3	4	5	6	7	8	9
sun	food	chip	god	people	space	file	drive	thank	team
de	arbor	clipper	believe	kill	government	window	win	please	player
objective	ann	encryption	faith	turk	bony	win	floppy	file	game
tu	thing	escrow	atheist	say	jake	run	hard	mail	hockey
font	disease	boston	hell	right	president	program	boot	advance	play
graphic	name	key	say	fire	center	directory	problem	anyone	season
value	mark	algorithm	atheism	attack	launch	manager	controller	duke	league
display	mi	machine	existence	turkey	book	server	monitor	hi	win
motif	question	privacy	belief	country	shuttle	buffalo	switch	know	ca
program	level	government	exist	war	moon	version	se	state	ranger

Table 17: Top ten words for the eight topics for the NMF model for the Emails dataset.

0	1	2	3	4	5	6	7
one	may	say	anonymous	health	use	privacy	key
say	address	go	use	year	law	file	bit
go	anonymous	president	service	disease	government	information	one
people	user	know	posting	child	encryption	network	planet
would	people	think	post	number	chip	technology	first
get	anonymity	work	user	report	technology	network	block
come	system	well	anon	medical	new	security	earth
see	right	take	server	state	clipper	mail	chip
like	identity	make	message	patient	would	public	message
could	many	key	version	study	key	policy	system

Table 22: Top ten words for the eight topics for the NMF model with tfidf weighting for the Emails dataset.

0	1	2	3	4	5	6	7
clipper	chip	phone	get	pat	space	government	key
encryption	device	trust	drug	bit	stratus	algorithm	bit
phone	encryption	government	go	access	mail	encryption	public
key	algorithm	security	want	net	post	patent	session
chip	use	president	fire	run	secret	secret	encrypt
system	clipper	privacy	good	give	news	people	escrow
escrow	voice	ensure	tell	thing	please	right	number
use	technology	care	system	two	world	public	serial
right	datum	omission	would	think	thank	law	chip
criminal	privacy	law	gun	go	year	say	block

Table 18: Top ten words for ten selected topics for the NMF model for the GitHub projects dataset.

0	1	2	3	4	5	6	7	8	9
type	ly	type	type	type	get	flow	flow	flow	flow
boost	get	flow	boost	flow	ly	direction	type	type	direction
get	qp	get	value	boost	type	ptr	direction	direction	type
value	type	name	get	get	qp	uint	name	name	ptr
size	vl	ptr	d	ptr	vl	name	ptr	boost	uint
result	wd	value	pp	ptr	wd	type	get	ptr	name
pp	d	op	i	direction	value	get	uint	get	get
string	value	uint	result	uint	set	count	d	uint	count
d	i	size	string	pp	ul	summary	i	pp	summary
set	err	direction	set	value	d	vtbl	ly	value	vtbl

Table 23: Top ten words for ten selected topics for the NMF model with tfidf weighting for the GitHub projects dataset.

0	1	2	3	4	5	6	7	8	9
candle	table	sql	vue	react	wallet	webpack	wallet	pyx	chart
shardingsphere	db	summary	texture	node	mut	plot	gl	wallet	sqlite
glew	column	sprite	webpack	model	std	torch	train	serie	torch
prop	client	player	xonsh	style	pub	kwargs	axis	err	price
price	i	span	tensor	state	ua	np	tx	tensor	tensor
trade	func	candle	neo	axis	ssh	train	chart	glew	exchange
xonsh	err	engine	vertex	font	prism	trino	vulkan	pub	tensor
apache	license	node	blockchain	value	orbitdb	numref	block	mut	market
webpacker	d	hljs	camera	event	onig	plt	sph	crate	mut
trading	np	puma	vec	get	plt	shiv	layer	crate	license

Table 19: Top ten words for ten selected topics for the NMF model for the Reuters dataset.

0	1	2	3	4	5	6	7	8	9
company	source	profit	currency	march	say	year	price	rise	barrel
say	oil	loss	exchange	week	billion	last	new	fall	crude
corp	official	net	market	rise	sale	one	pact	year	west
unit	share	share	fed	price	end	price	rubber	month	raise
business	crude	note	foreign	compare	gas	say	world	adjust	grade
sell	tax	include	mark	quarter	expect	growth	market	index	price
expect	last	include	rate	bill	total	market	reference	output	posting
operation	could	company	system	gain	early	economic	conference	production	oil
year	export	sale	bank	first	rise	expect	may	seasonally	contract
product	plan	gain	west	drop	low	economy	month	say	petroleum

Table 24: Top ten words for ten selected topics for the NMF model with tfidf weighting for the Reuters dataset.

0	1	2	3	4	5	6	7	8	9
give	sugar	sugar	currency	billion	acquisition	output	franklin	buy	mine
corp	rubber	white	baker	net	acquire	production	corp	share	dome
assistance	yr	rebate	dollar	rise	common	china	fund	common	offer
band	per	trader	stability	mark	complete	chip	income	stake	bid
class	oil	intervention	exchange	yen	tender	month	insure	exchange	south
public	market	export	franklin	year	pacific	life	board	march	debt
go	extraordinary	declare	west	surplus	stock	sugar	group	group	gold
help	consolidated	tender	industrial	deposit	company	cotton	march	agree	miner
late	cattle	cargo	finance	end	hotel	period	tax	security	copper
shortage	gas	dividend	treasury	reserve	prime	lynch	bell	commission	mining

Table 20: Top ten words for ten selected topics for the NMF model for the Seven Categories dataset.

0	1	2	3	4	5	6	7	8	9
author	war	force	share	force	charge	water	energy	share	vector
book	cash	motion	call	use	wave	surface	particle	light	cell
write	soviet	two	chromosome	time	field	electric	per	united	use
movement	also	body	capital	time	electric	air	time	new	also
include	united	state	figure	power	point	pressure	mass	world	system
new	state	change	two	memory	time	plant	share	call	light
many	government	law	issue	war	two	ocean	point	war	organism
soviet	union	velocity	become	friction	direction	call	work	account	time
national	new	acceleration	year	would	give	due	potential	use	two
country	become	fig	population	china	surface	movement	surface	also	animal

Table 25: Top ten words for ten selected topics for the NMF model with tfidf weighting for the Seven Categories dataset.

0	1	2	3	4	5	6	7	8	9
plate	cell	cash	water	share	united	debenture	charge	wave	specie
motion	inequality	decimal	pressure	nucleus	state	issue	magnetic	light	water
body	plant	share	share	nuclear	soviet	party	field	electron	wave
particle	lesson	digit	charge	energy	pressure	premium	force	speed	plant
system	review	number	segment	equity	war	company	energy	acceleration	energy
capacitor	instruction	percent	surface	neutron	new	government	electric	velocity	light
rigid	chromosome	pay	wire	capital	return	share	vector	cash	fraction
force	stage	round	company	decay	economic	allotment	mass	motion	organism
axis	figure	cheque	tissue	reactor	president	election	point	object	diversity
rotation	transport	point	blood	application	world	reserve	surface	statement	ecosystem

Table 26: Top ten words for ten selected topics for the BERT model for the 20 Newsgroups dataset.

0	1	2	3	4	5	6	7	8	9
file	dog	food	patient	nuclear	space	president	jake	marriage	buffalo
drive	chase	eat	disease	cool	launch	job	bony	marry	ra
state	bike	diet	doctor	tower	orbit	secretary	slave	married	ye
computer	driveway	seizure	medical	plant	moon	package	true	ceremony	god
want	ride	sensitivity	pain	water	satellite	administration	opinion	wife	paradise
right	springer	taste	cancer	steam	henry	russia	russia	husband	father
window	questor	superstition	treatment	uranium	lunar	senator	bush		salvation
thing	dod	corn	infection	reactor	mission	senator	slavery	couple	child
problem	attack	restaurant	health	fossil	shuttle	summer	prison	eye	thou
host	boom	dyer	yeast	temperature	earth	press	aid	commitment	mormon

Table 27: Top ten words for the eight topics for the BERT model for the Emails dataset.

0	1	2	3	4	5	6	7
gun	encryption	tap	bit	secrecy	trial	apple	agency
state	entitle	betel	sequence	agora	thousand	trust	safeguard
come	effective	traffic	chunk	rain	microsecond	credibility	escrow
right	threaten	prefix	actual	den	engine	secret	expedient
party	unbreakable	blank	split	repository	skipjack	forever	congress
war	safety	encryption	console	door	hypothesize	bu	funded
host	illegal	accuse	competence	variant	arithmetic	assurance	overtly
want	privacy	decode	random	security	brute	unsound	federal
law	saying	unauthorized	partial	instant	steven	steal	fund
world	outright	technique	randomly	mnemonic	million	connect	assurance

Table 28: Top ten words for ten selected topics for the BERT model for the Reuters dataset.

0	1	2	3	4	5	6	7	8	9
bank	mar	money	canada	sale	net	sale	corp	sale	net
rate	park	assistance	dome	net	year	net	net	net	roto
sale	net	shortage	trade	diluted	renaissance	diluted	rev	mercantile	yr
corp	zone	market	bank	dilute	pembina	year	odeon	crain	lighting
quarter	faulty	band	shell	corp	oe	vanguard	year	potlatch	dunk
stock	favorite	fed	rise	gem	charming	suspension	basic	servo	analogic
new	favorably	bank	billion	sexton	tempo	note	wallpaper	electromagnetic	dorn
end	favorable	reserve	air	ply	vanguard	backlog	pyro	whirlpool	napa
note	favor	forecast	fall	seam	foremost	end	sidy	translation	rooter
sell	fault	today	royal	burr	kay	rivet	gas	fuller	morse

Table 29: Top ten words for seven selected topics for the BERT model for the Seven Categories dataset.

| 0	1	2	3	4	5	6
space	cash	cache	motion	wind	united	plant
population	debenture	bit	particle	tide	soviet	organism
mission	account	address	fig	air	war	cell
satellite	share	register	magnetic	ocean	president	animal
pregnancy	company	processor	electric	warm	party	specie
moon	allotment	memory	wave	condensation	government	tissue
craft	purchase	computer	electron	cloud	union	blood
launch	capital	page	mass	cyclone	country	biology
exploration	balance	datum	velocity	tropical	political	acid
orbit	pay	clock	direction	circulation	military	protein
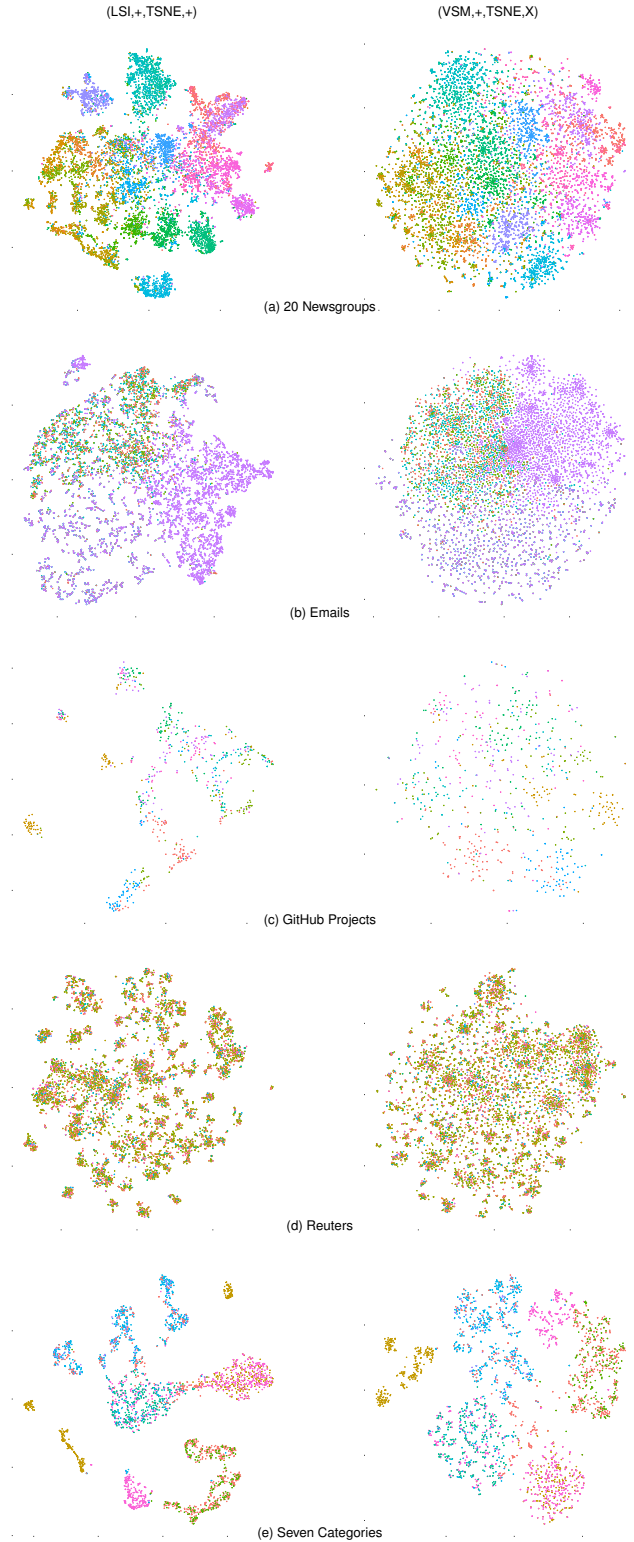
Figure 1: Layout examples for LSI and VSM with applied t-SNE dimensionality reduction for the five datasets.