

TEMPORALLY COHERENT VISUALISATION OF TIME-DEPENDENT DATA

Anonymous authors

Paper under double-blind review

ABSTRACT

Dimension reduction algorithms aim to embed high-dimensional datasets into a low-dimensional space in such a way that important structural properties, such as clusters and manifolds, are preserved. Most such methods are designed for static data, and naively applying them to time-dependent data can lead to unstable embeddings which do not meaningfully capture the temporal evolution of the data. In this paper, we propose a new variant of the t-SNE algorithm for time-dependent data, TC-tSNE (Temporally Coherent t-SNE) in which an extra term is added to the cost function to promote temporal coherence: the notion that a data point which has a similar position in two time frames should be embedded to similar positions at those times. Importantly, this notion captures temporal similarities over the entire time domain and can therefore capture long-range temporal patterns, not just local ones. We demonstrate the effectiveness of our method for visualising dynamic network embedding, and we evaluate our method on six benchmark datasets using a collection of metrics, which capture the structural quality and the temporal coherence of the embeddings. We compare our method with existing dynamic visualisation algorithms and find that it performs competitively.

1 INTRODUCTION

Exploratory analysis of complex datasets is a fundamental task across science, industry and government, and often starts with data visualisation. Good visualisation algorithms can reveal the important structural properties of a dataset, such as clusters, manifolds and outliers, which may not be immediately apparent from the raw data. In recent years, neighbor-embedding algorithms such t-SNE (Van der Maaten & Hinton, 2008) and UMAP (McInnes et al., 2018), and their many variants, have proved enormously successful for this task, for example becoming a standard part of the scientific practice in genomics (Kobak & Berens, 2019), neuroscience (Dimitriadis et al., 2018), molecular biology (Li et al., 2017) and many other fields.

In many domains, datasets have a natural temporal component, and data points change and evolve through time. Examples include collaborative document editing, neural imaging, dynamic networks and video and sound recording. One might hope that a good visualisation algorithm could also reveal the important temporal patterns in such datasets, such as trends, seasonalities, changepoints and reversals to previous states.

While there is a highly-developed literature on data visualisation for static data, comparatively little work has focused on data visualisation algorithms for time-dependent data. As well as representing the spatial properties of the data, such algorithms face the additional challenge of faithfully representing its temporal evolution. We refer to this desideratum as *temporal coherence*.

In a temporally coherent embedding, embedded positions should only move when the corresponding data points move; clusters in the embedding space should merge and split as they do in the data; and should a data point reverse to previous position, the corresponding embedded position should reverse too. Embeddings which fail to achieve this goal may mislead a user into thinking that a change has happened when it has not, or that two data points have switched places when they have not. Simultaneously achieving high spatial quality, and maintaining temporal coherence is a challenging task, and most existing algorithms seem to favour one goal or the other. In this paper, we present a strategy which achieves both.

2 EXISTING STRATEGIES FOR TIME-DEPENDENT DATA VISUALISATION

To see why this is such a challenging task, we present six existing strategies and describe their failure modes. The first four are simple plug-in adjustments to existing base methods, while the final two adjust the cost function by adding additional terms which aim to promote temporal coherence.

Independent. The simplest possible strategy for time-dependent data visualisation is to take a static embedding algorithm and apply it to each time frame independently. Unfortunately, this tends to lead to highly unstable embeddings. To see why this is, consider an embedding algorithm whose objective function depends only on interpoint distances in the data and embedding spaces, such as t-SNE or UMAP. In this case, small changes in the initialisation, data and any randomness into the optimisation procedure can result in dramatic rotations in the embedding space. PCA suffers the same fate due to the ambiguity of the signs of eigenvectors and the ordering of the dimensions by eigenvalue. One benefit of this strategy is that it tends to produce embeddings with the highest spatial quality.

Global. On the other end of the spectrum is the global strategy, in which all timeframes and combined into a single dataset and embedded as one. By construction, this strategy will produce temporally coherent embeddings and it forms the basis of a variety of existing dynamic visualisation algorithms (Hu et al., 2010; Fujiwara et al., 2018; 2020; Crnovrsanin et al., 2009). In the context of dynamic network embedding, a recent line of works apply a global embedding strategy to obtain temporally coherent variants of spectral and skip-gram-based graph embedding algorithms (Gallagher et al., 2021; Modell et al., 2023; Davis et al., 2023). Despite this, positioning each data point relative to all other data points at *all* time points can lead to poor spatial quality which can be particularly prevalent in the context of visualisation, where one has access to at most three dimensions.

Aligned. A simple strategy to alleviate the problems of the independent embedding strategy is to apply a post-hoc transformation to the embeddings at each timeframe to align them with the previous timeframes. This is typically achieved using sequential orthogonal Procrustes alignments (Schönemann, 1966). This strategy has been used in a number time-dependent embedding algorithms including visualisation (Crnovrsanin et al., 2009), temporal analysis using word embeddings (Kim et al., 2014; Yao et al., 2018; Szymanski, 2017; Kulkarni et al., 2015) and temporal network embeddings (Singer et al., 2019; Zhou et al., 2019; Cape, 2021). The alignment strategy makes the implicit assumption that the data evolves smoothly, and that temporally adjacency timeframes are similar. Even if this holds, the strategy also suffers from drift in the long-term, and if it does not, it can be highly unstable. For these reasons, long-range temporal relationships are not reliable.

Continuous. Another strategy which applies to optimisation-based methods is to sequentially optimise each timeframe, initialing the optimisation of the embedding for timeframe t with the embedding computed for timeframe $t - 1$. We refer to this as the continuous strategy. This strategy has been employed in the context of temporal word and network embeddings (Kim et al., 2014; Zhou et al., 2019; Szymanski, 2017), however it unfortunately suffers from many of the same drawbacks as the alignment strategy.

Velocity-penalised. This strategy involves simultaneously optimising the embeddings for each time frame subject to an additional penalty term in the cost function which penalises large movements in the latent space. We refer to this approach as the *velocity penalised* approach. Rauber et al. (2016) apply this approach to the t-SNE algorithm, and optimise the cost function

$$C = \sum_{t=1}^T C^t + \frac{\lambda}{2n} \sum_{i=1}^n \sum_{t=2}^T \|y_i^t - y_i^{t-1}\|^2,$$

where C^t is the usual t-SNE cost function for timeframe t , and y_i^t is the embedded position of data point i at time t . This strategy has also been applied in the context of temporal graph layouts (Xu et al., 2013; Leydesdorff & Schank, 2008) and temporal network and word embeddings (Yao et al., 2018; Singer et al., 2019; Zhou et al., 2019; Rastelli & Corneli, 2023). The strategy has two distinct failure modes which can prevent it from maintaining temporal: firstly, if two data points meet and then diverge, the cost function will “forget” where they came from, and secondly, if there are abrupt

changes in the data, the velocity penalisation term will prevent the embeddings from capturing this change.

Guided. In this strategy, introduced for time-dependent t-SNE embeddings by Vernier et al. (2021), the embeddings for each timeframe are optimised independently with respect to their usual cost function plus a shared global cost function, which guides the placement of the points and encourages temporal coherence. Different forms of this guide lead to different variants of the algorithm. One variant, Landmark Dynamic t-SNE, chooses m landmark points and adds either a global PCA, or global asymmetric t-SNE penalty to the objective for these points. Another variant, Principal Component Dynamic t-SNE guides embeddings towards a global PCA.

3 OUR STRATEGY

One of the major drawbacks of the aforementioned strategies for time-dependent data visualisation, with the exception of the global and guided strategies, is that they only aim to promote temporal coherence over short time-scales. However, many important exploratory analyses require temporal coherence over long time-scales. In these contexts, these strategies are inappropriate and could lead to misleading inferences. In this section, we will informally outline our strategy, which is fundamentally different from any of those outlined above. Our strategy can be applied to any neighbor-embedding method, and we will outline informally in this section. In the next section, we will formally describe an variant of our strategy based on t-SNE, which we called TC-tSNE (Temporally Coherent t-SNE).

3.1 STRUCTURAL AND TEMPORAL COHERENCE

We will assume that our dataset is made up of n datapoints, each observed over T timeframes, and we will denote the i th datapoint at the t th timeframe by x_i^t . Similarly, we will let y_i^t denote the embedding of the i th datapoint at timeframe t . A neighbor-embedding objective function is designed so that points which are nearby in the data space and positioned close to each other in the embedding space. If an embedding achieves this goal for every timeframe, informally we will say that it is *spatially coherent*. In symbols, an embedding is *spatially coherent* if and only if

$$\|x_i^t - x_j^t\| \text{ is small} \iff \|y_i^t - y_j^t\| \text{ is small} \quad \text{for all } i, j \in [n], t \in [T]. \quad (1)$$

Analogously, we will define the (informal) notion of *temporal coherence* as follows: we say an embedding is *temporally coherence* if every point that has similar positions in the data space at two different timeframes has similar positions in the embedding space at those timeframes. In symbols, we say an embedding is *temporally coherent* if and only if

$$\|x_i^t - x_i^s\| \text{ is small} \iff \|y_i^t - y_i^s\| \text{ is small} \quad \text{for all } i \in [n], t, s \in [T]. \quad (2)$$

Importantly, our notion of temporal coherence covers *all* time ranges, both long and short.

Many of the aforelisted strategies, such as the aligned, continuous and velocity-penalised strategies *do not* aim to satisfy this notion of temporal coherence. Instead, they only aim to satisfy what we will refer to as *local temporal coherence*, where $s = t - 1$.

3.2 A STRUCTURAL - TEMPORAL DUALITY

One might notice that, upon swapping the datum and time indicies, the temporal coherence definition (2) becomes equivalent to the structural coherence definition (1). This motivates us to consider cost functions of the following form:

$$C := \frac{1}{T} \sum_{t=1}^T C^t + \frac{\lambda}{n} \sum_{i=1}^n \tilde{C}_i \quad (3)$$

where C^t denotes a cost function of a chosen base method with respect to the dataset $\{x_1^t, \dots, x_n^t\}$, and \tilde{C}_i denotes a cost function of a chosen base method with respect to the dataset $\{x_i^1, \dots, x_i^T\}$. We will refer to these cost functions as the spatial and temporal costs respectively. Notice that each

162 data points (and each embedded point) appears in exactly one spatial cost term and one temporal
 163 cost term and therefore the gradients have the simple form:

$$164 \frac{\partial C}{\partial y_i^t} = \frac{1}{T} \cdot \frac{\partial C^t}{\partial y_i^t} + \frac{\lambda}{n} \cdot \frac{\partial \tilde{C}_i}{\partial y_i^t}.$$

165
 166
 167 One key advantage of this strategy is that it can be plugged into *any* neighbor embedding algorithm
 168 with little modification to the code. One simply needs to compute the gradients using the base
 169 method for the relevant datapoints. Computationally, each of our gradient calculations involves
 170 calculating the base-method gradient on T datasets of n points and n datasets if T points.
 171

172 4 TC-TSNE: TEMPORALLY COHERENT T-SNE

173
 174 In this section, we formally describe a specific version of our strategy which we have found to
 175 work extraordinarily well in practice. For the spatial cost function we use the t-SNE (t-Distributed
 176 Stochastic Neighbor Embedding) cost function, and for the temporal cost function, we use the sym-
 177 metric Stochastic Neighbour Embedding (SNE) (Hinton & Roweis, 2002) cost function. The reason
 178 for using SNE over t-SNE for the temporal costs, is that t-SNE induces clustering which can be
 179 desirable in the spatial domain, but is not necessarily meaningful in the temporal domain. In addi-
 180 tion to the parameter λ , the spatial and temporal qualities of the will be controlled by user-specified
 181 spatial and temporal perplexity parameters, μ and $\tilde{\mu}$ respectively.

182 For completeness, we will briefly describe the temporal and spatial costs and gradients derived from
 183 the SNE and t-SNE objectives.
 184

185 **Temporal gradients (SNE).** The SNE objective aims to match the conditional probabilities that
 186 a data point would select another as its neighbour if neighbours were picked in proportion to their
 187 probability density under a Gaussian centered at it, in the data space and the embedding space.
 188 For each $i \in [n]$, we define a joint probability distribution \tilde{P}_i over the data point i at all pairs of
 189 timeframes $\{(x_i^t, x_i^s)\}_{t \neq s}$ via

$$190 \tilde{p}_i^{ts} := \frac{\tilde{p}_i^{t|s} + \tilde{p}_i^{s|t}}{2} \quad \text{where} \quad \tilde{p}_i^{s|t} = \frac{\exp(-\|x_i^t - x_i^s\|^2 / 2\tilde{\sigma}_i^t)}{\sum_{r \neq t} \exp(-\|x_i^t - x_i^r\|^2 / 2\tilde{\sigma}_i^t)}$$

191 where $\tilde{\sigma}_i^t$ is a parameter which is chosen such that $\tilde{\mu} = 2^{\sum_s \tilde{p}_i^{s|t} \log_2 \tilde{p}_i^{s|t}}$, where $\tilde{\mu}$ is a user-specified
 192 *temporal perplexity* parameter.
 193

194 We define a joint distribution \tilde{Q}_i over the embedding of datum i at all pairs of timeframes
 195 $\{(y_i^t, y_i^s)\}_{t \neq s}$ via

$$196 \tilde{q}_i^{ts} := \frac{\exp(-\|y_i^t - y_i^s\|^2)}{\sum_{r \neq u} \exp(-\|y_i^r - y_i^u\|^2)}.$$

197 The temporal cost for the i th datum is then given by the Kullback-Leibler divergence between \tilde{P}_i
 198 and \tilde{Q}_i . That is

$$199 \tilde{C}_i := D_{\text{KL}}(\tilde{P}_i \| \tilde{Q}_i) = \sum_{t \neq s} \tilde{p}_i^{ts} \log \frac{\tilde{p}_i^{ts}}{\tilde{q}_i^{ts}},$$

200 the gradients of which is given by

$$201 \frac{\partial \tilde{C}_i}{\partial y_i^t} = 4 \sum_{s \neq t} (\tilde{p}_i^{ts} - \tilde{q}_i^{ts})(y_i^t - y_i^s).$$

202
 203 **Spatial gradients (t-SNE).** For the spatial costs, we employ the t-SNE objective, which induces
 204 clustering in the embedding space by matching Gaussian densities in the data space with heavier
 205 tailed Student's t-distribution densities in the embedding space. As in SNE, for each $t \in [T]$, we
 206 define a joint probability distribution \mathcal{P}^t , over all pairs of data points at timeframe t , $\{(x_i^t, x_j^t)\}_{i \neq j}$
 207 via

$$208 p_{ij}^t := \frac{p_{i|j}^t + p_{j|i}^t}{2} \quad \text{where} \quad p_{j|i}^t = \frac{\exp(-\|x_i^t - x_j^t\|^2 / 2\sigma_i^t)}{\sum_{r \neq t} \exp(-\|x_i^t - x_j^t\|^2 / 2\sigma_i^t)}$$

216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269

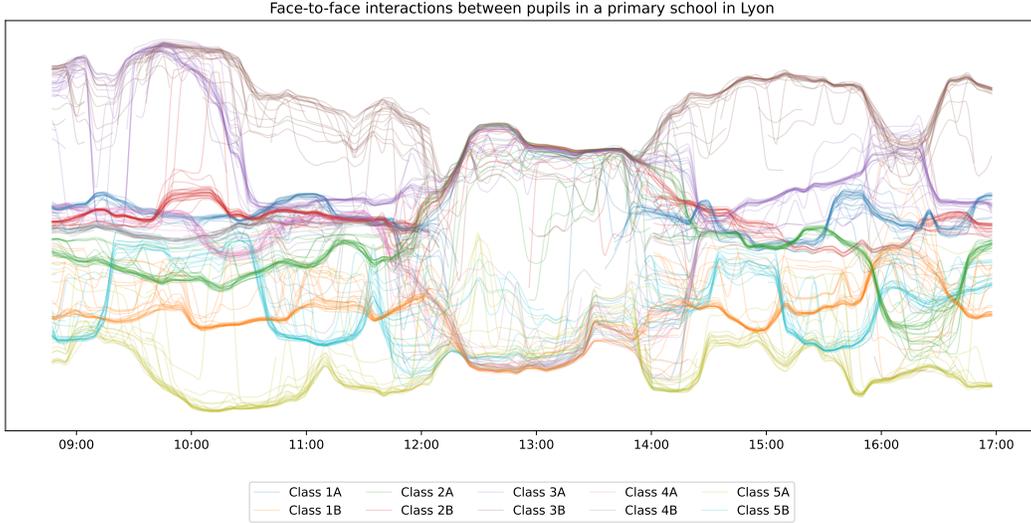


Figure 1: One-dimensional TC-tSNE applied to UASE dynamic network embeddings of a face-to-face interaction network of pupils at a primary school in Lyon. Colors correspond to the class memberships of the pupils.

where σ_i^t is a parameter which is chosen such that $\mu = 2^{\sum_s p_{j|i}^t \log_2 p_{j|i}^t}$, where $\tilde{\mu}$ is a user-specified *temporal perplexity* parameter.

The joint distribution Q^t over all pairs of embeddings at timeframe t , $\{(y_i^t, y_j^t)\}_{i \neq j}$ is defined via

$$q_{ij}^t := \frac{(1 + \|y_i^t - y_j^t\|^2)^{-1}}{\sum_{k \neq l} (1 + \|y_k^t - y_l^t\|^2)^{-1}}$$

The spatial cost for the t th timeframe is then given by the Kullback-Leibler divergence between \mathcal{P}^t and Q^t . That is

$$C^t := D_{\text{KL}}(\mathcal{P}^t \| Q^t) = \sum_{t \neq s} p_{ij}^t \log \frac{p_{ij}^t}{q_{ij}^t},$$

and the gradients are given by

$$\frac{\partial C^t}{\partial y_i^t} = 4 \sum_{j \neq i} (p_{ij}^t - q_{ij}^t) (y_i^t - y_j^t) (1 + \|y_i^t - y_j^t\|^2)^{-1}.$$

Optimisation. In our experiments, we optimise (3.2) using the vanilla gradient descent with momentum algorithm described in the original t-SNE paper (Van der Maaten & Hinton, 2008), and apply the early exaggeration strategy. However, our gradients can be easily plugged into accelerated algorithms such as Barnes-Hut (Van Der Maaten, 2014) and Fast-Fourier-Transform-accelerated algorithms (Linderman et al., 2017; 2019) which scale significantly better than the original implementation. For brevity, we refer the reader to the references for details of optimisation strategies.

5 CASE STUDY: VISUALISATION OF DYNAMIC NETWORK EMBEDDINGS

A key motivation for this work was to create a tool for visualising the evolutions of dynamic networks. A recent line of works have developed spectral embedding algorithms for dynamic networks with temporal coherence guarantees of the kind discussed here. By nature, these are *linear* dimension-reduction methods, and as such, often require a moderate number of dimensions to capture all the salient information in the data. For the datasets we have studied, this is typically

270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323

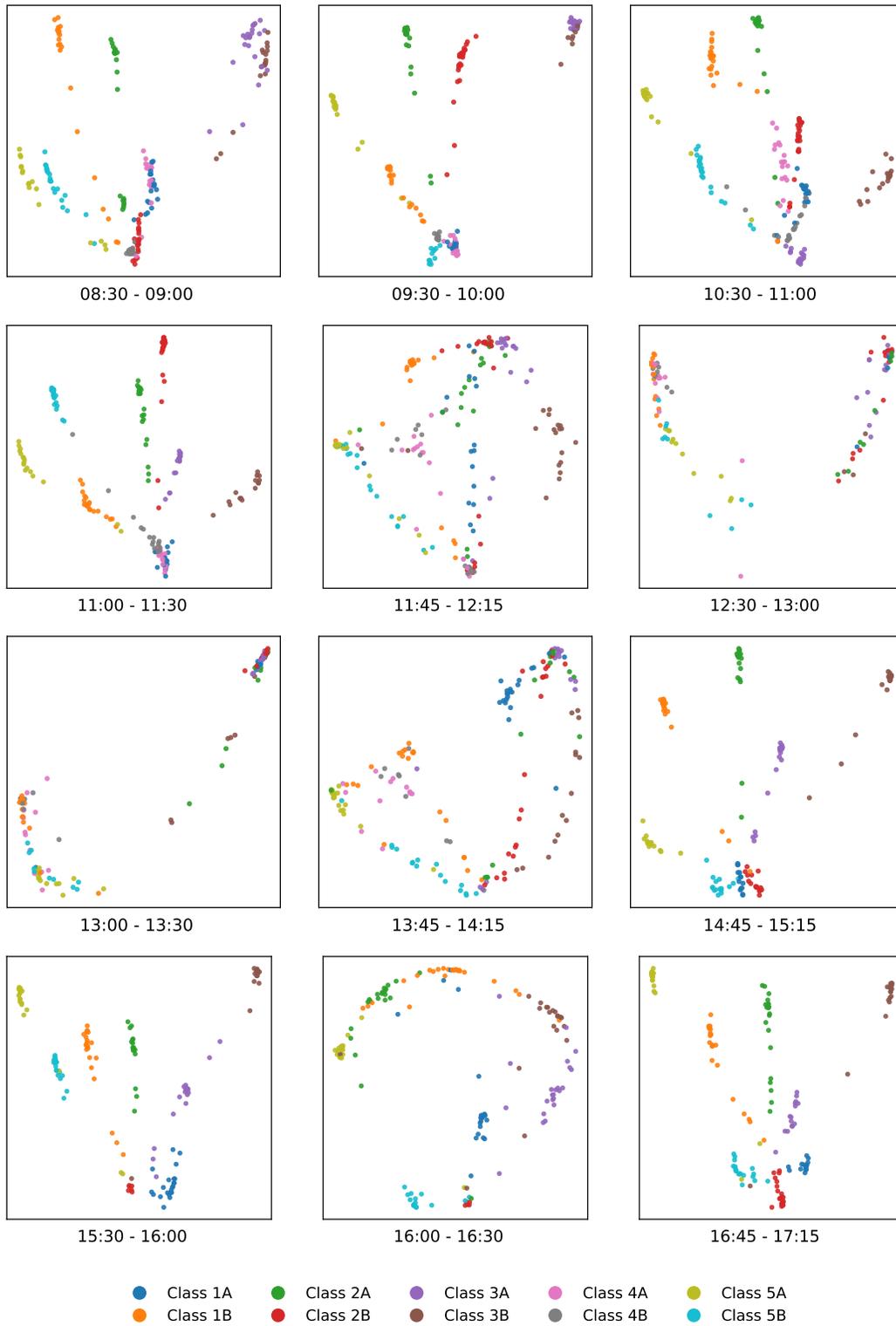


Figure 2: Two-dimensional TC-tSNE applied to UASE dynamic network embeddings of a face-to-face interaction network of pupils at a primary school in Lyon. Colors correspond to the class memberships of the pupils.

324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377

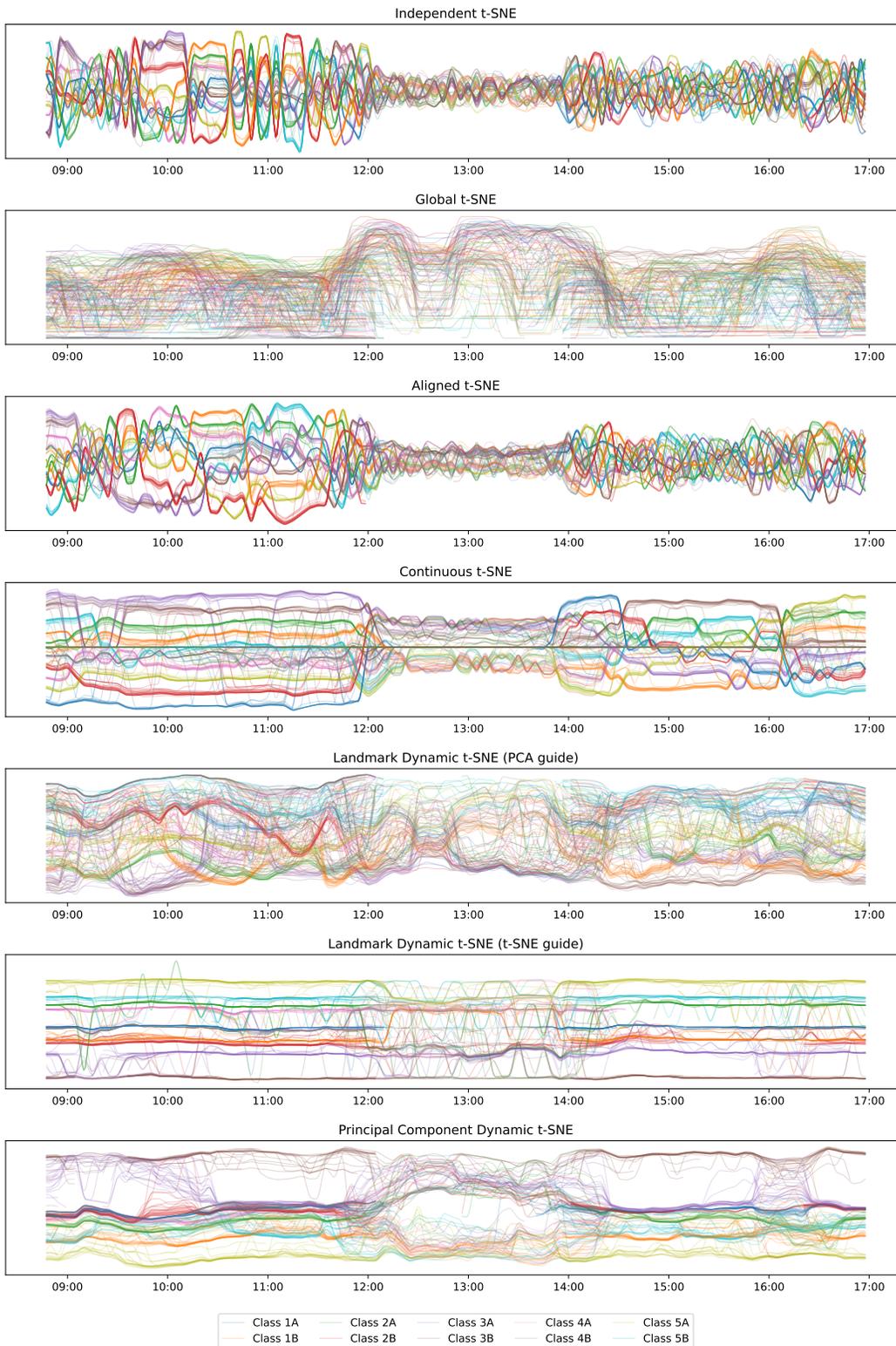


Figure 3: Some alternative time-dependent visualisation algorithms applied the data described in Section 5.

378 somewhere between 5 and 100 dimensions — certainly more than the three available to the human
379 eye.

380 Both of the aforementioned papers study a dataset of face-to-face interactions between pupils at a
381 primary school in Lyon and compute time-evolving embeddings of the pupils based on their interac-
382 tion patterns. Details of the study and data collection can be found in Stehlé et al. (2011). Gallagher
383 et al. (2021) plot the first two coordinates of the embeddings and Modell et al. (2023) individually
384 perform one-dimensional global PCA to subsets of the pupils corresponding to year groups, which
385 they plot against a time axis. Both of these approaches are unsatisfactory, and fail to clearly present
386 the rich underlying structure in the data. TC-tSNE provides a solution.

387 To demonstrate this, for a given day, we first construct a sequence of 100 graphs obtained by taking
388 a 30-minute sliding window over the school day in steps of 5 minutes, and placing an edge in
389 the relevant graph if there was a physical interaction recording between two students during the
390 time window. We then apply the Unfolded Adjacency Spectral Embedding algorithm of Gallagher
391 et al. (2021) into 9 dimensions, which we normalise to lie on the unit sphere (following the Real
392 Data section in that paper). When a pupil has no interactions in a timeframe we treat that point
393 as missing. We apply TC-tSNE into one-dimension with the λ parameter set to 0.3, the spatial
394 perplexity parameter set to $\mu = 30$ and the temporal perplexity parameter set to $\tilde{\mu} = 5$.

395 Figure 1 shows the TC-tSNE embedding plotted against time, where we have interpolated time-
396 frames piecewise-linearly and coloured lines according to the class to which the pupil belongs. Even
397 without knowing any more about the data, one can clearly see the separation of classes during the
398 morning and afternoon periods, and the merging of classes into two groups during the lunch period.

399 Figure 2 shows a collection of timeframes from a two-dimensional TC-tSNE applied to the same
400 data. The plots clearly present the separation of pupils during class times, mingling during break
401 times, and the return of students to their classes.

402 For comparison, Figure 3 shows one-dimensional visualisations of these network embeddings using
403 the t-SNE based methods described in Section 2. We were unable to apply the velocity-penalised
404 dynamic t-SNE algorithm of Rauber et al. (2016) since it cannot handle missing values in the data.
405 Where there were hyperparameters to tune, we chose the parameters which looks best by eye, how-
406 ever none of these existing methods show the same level of fidelity as TC-tSNE.

407 As expected, Independent t-SNE has the worst temporal coherence and Global t-SNE has the worst
408 spatial quality of all the methods. Aligned t-SNE does a slightly better job than Independent t-SNE
409 at maintaining some local temporal coherence, although the improvement is minor. Continuous t-
410 SNE does a better job at balancing temporal coherence and spatial quality, although many of the
411 classes return to different positions after the lunch break. Both Landmark Dynamic t-SNE methods
412 (Vernier et al., 2021) perform poorly in terms of spatial quality, with neither clearly showing the
413 separation of pupils into two groups during the lunch break. Principal Component Dynamic t-SNE
414 (Vernier et al., 2021) performs well in terms of temporal coherence, although the spatial clustering
415 is much less clear than in the TC-tSNE embedding.

417 6 QUANTITATIVE EVALUATION

418 In this section, we present a quantitative comparison of our method, TC-tSNE, against some of
419 the methods described in Section 2.

420
421
422 **Methods.** We compare to independent, global and aligned PCA and t-SNE, continuous t-SNE,
423 the velocity penalized dynamic t-SNE algorithm (D-tSNE) of Rauber et al. (2016), the Landmark
424 Dynamic t-SNE (LD-tSNE) algorithm (with PCA and t-SNE guides) and Principal Component Dy-
425 namic t-SNE (PCD-tSNE) algorithms of Vernier et al. (2021). For all t-SNE based methods, we
426 used a perplexity of either 10, 20 or 30, depending on the dataset which was chosen by eye before
427 the experiment using independent t-SNE embeddings.

428
429 **Metrics.** To evaluate the methods we employ five spatial metrics introduced in Espadoto et al.
430 (2019) and Vernier et al. (2021) and we introduce a new temporal metric which measures temporal
431 coherence across all time-ranges. We employ four metrics which measure the local quality of the
embeddings: *Neighborhood preservation*, *Neighborhood hit*, *Trustworthiness* and *Continuity*. Each

432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485

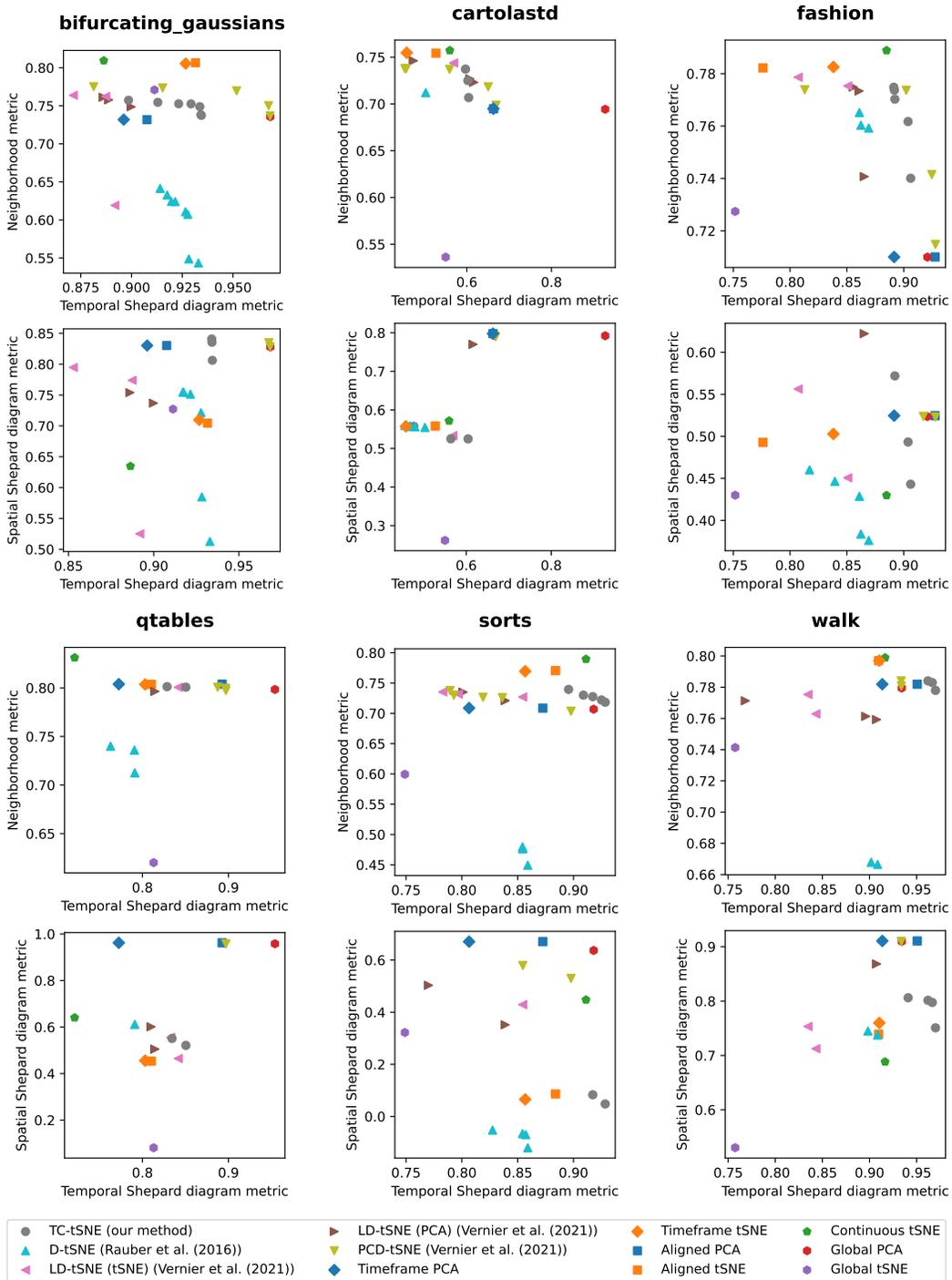


Figure 4: Spatial vs temporal coherence metrics for a variety of time-dependent visualisation algorithms on a set of benchmark datasets.

of these metrics is parametrised by a number-of-nearest neighbors. Following Vernier et al. (2021), we compute the metrics for a range of 20 values of k ranging from 1% to 20% of the data (except Neighborhood hit, where to take values of k from 0.25% to 5% of the data), and record the average. For a given metric and dataset, we report the average of these four metrics, each summed over all data points and timeframes. We call the value the neighborhood metric. In addition, we consider spatial Shepard diagram metrics, which measure a correlation coefficient of a scatterplot of interpoint distances in the data space against interpoint distances in the embedding space. We do this using Pearson correlation, Spearman rank and Kendall Tau coefficients, and report the average of these three.

To measure *local temporal coherence*, Vernier et al. (2021) introduced a temporal version of the Shepard diagram metric which measures the correlation of the scatter plot of interpoint distances between individual datapoints at neighboring timeframes. We introduce a variant of this metric which measures the correlation of the scatter plot of interpoint distances between individual datapoints across *all* timeframes. We will refer to this metric as the *temporal Shepard diagram metric*.

Datasets. For our comparison, we use six datasets. The first “bifurcating_gaussians” is a synthetic dataset of 250 Gaussian blobs which are initially separated into five groups, merge, remain merged for some time before returning to their original positions. The other five datasets, “cartolastd”, “fashion”, “qttables”, “sorts” and “walk” and taken from Vernier et al. (2020) and descriptions of them can be found there.

Results. For each dataset, Figure 4 shows the temporal Shepard diagram metric, plotted against the neighborhood metric and the spatial Shepard diagram metric for each of the methods. For methods with hyperparameters, we computed the metrics for a variety of parameter choices and plotted the metrics for all the choices which were not uniformly dominated by another hyperparameter choice on both the spatial and temporal axes.

For most datasets, our method performs competitively sometimes achieving the best temporal and spatial metrics, and is never among the worst.

7 CONCLUSION

In this paper, we have presented a new strategy for temporally coherent visualisation of time-dependent data. We have presented a specific variant of this strategy, TC-tSNE, which we have shown to perform excellently at the task of visualising a dynamic network embedding. Our algorithmic framework opens new possibilities for exploratory analysis of time-dependent datasets, and we are excited to see how our method can be used in the applied sciences. In future work, we hope to apply our strategy to other neighbor-embedding methods such as UMAP, and to understand how our ideas can be applied for network and word embeddings.

REFERENCES

- Joshua Cape. Spectral analysis of networks with latent space dynamics and signs. *Stat*, 10(1):e381, 2021.
- Tarik Crnovrsanin, Chris Muelder, Carlos Correa, and Kwan-Liu Ma. Proximity-based visualization of movement trace data. In *2009 IEEE symposium on visual analytics science and technology*, pp. 11–18. IEEE, 2009.
- Ed Davis, Ian Gallagher, Daniel John Lawson, and Patrick Rubin-Delanchy. A simple and powerful framework for stable dynamic network embedding. *arXiv preprint arXiv:2311.09251*, 2023.
- George Dimitriadis, Joana P Neto, and Adam R Kampff. t-sne visualization of large-scale neural recordings. *Neural computation*, 30(7):1750–1774, 2018.
- Mateus Espadoto, Rafael M Martins, Andreas Kerren, Nina ST Hirata, and Alexandru C Telea. Toward a quantitative survey of dimension reduction techniques. *IEEE transactions on visualization and computer graphics*, 27(3):2153–2173, 2019.

- 540 Takanori Fujiwara, Jianping Kelvin Li, Misbah Mubarak, Caitlin Ross, Christopher D Carothers,
541 Robert B Ross, and Kwan-Liu Ma. A visual analytics system for optimizing the performance of
542 large-scale networks in supercomputing systems. *Visual Informatics*, 2(1):98–110, 2018.
- 543
544 Takanori Fujiwara, Naohisa Sakamoto, Jorji Nonaka, Keiji Yamamoto, Kwan-Liu Ma, et al. A visual
545 analytics framework for reviewing multivariate time-series data with dimensionality reduction.
546 *IEEE transactions on visualization and computer graphics*, 27(2):1601–1611, 2020.
- 547 Ian Gallagher, Andrew Jones, and Patrick Rubin-Delanchy. Spectral embedding for dynamic net-
548 works with stability guarantees. *Advances in Neural Information Processing Systems*, 34:10158–
549 10170, 2021.
- 550
551 Geoffrey E Hinton and Sam Roweis. Stochastic neighbor embedding. *Advances in neural informa-
552 tion processing systems*, 15, 2002.
- 553
554 Yueqi Hu, Shuangyuan Wu, Shihong Xia, Jinghua Fu, and Wei Chen. Motion track: Visualizing
555 variations of human motion data. In *2010 IEEE Pacific Visualization Symposium (PacificVis)*, pp.
556 153–160. IEEE, 2010.
- 557 Yoon Kim, Yi-I Chiu, Kentaro Hanaki, Darshan Hegde, and Slav Petrov. Temporal analysis of
558 language through neural language models. *arXiv preprint arXiv:1405.3515*, 2014.
- 559
560 Dmitry Kobak and Philipp Berens. The art of using t-sne for single-cell transcriptomics. *Nature
561 communications*, 10(1):5416, 2019.
- 562
563 Vivek Kulkarni, Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. Statistically significant detection
564 of linguistic change. In *Proceedings of the 24th international conference on world wide web*, pp.
565 625–635, 2015.
- 566
567 Loet Leydesdorff and Thomas Schank. Dynamic animations of journal maps: Indicators of structural
568 changes and interdisciplinary developments. *Journal of the American Society for Information
569 Science and Technology*, 59(11):1810–1818, 2008.
- 570
571 Wentian Li, Jane E Cerise, Yaning Yang, and Henry Han. Application of t-sne to human genetic
572 data. *Journal of bioinformatics and computational biology*, 15(04):1750017, 2017.
- 573
574 George C Linderman, Manas Rachh, Jeremy G Hoskins, Stefan Steinerberger, and Yuval Kluger.
575 Efficient algorithms for t-distributed stochastic neighborhood embedding. *arXiv preprint
576 arXiv:1712.09005*, 2017.
- 577
578 George C Linderman, Manas Rachh, Jeremy G Hoskins, Stefan Steinerberger, and Yuval Kluger.
579 Fast interpolation-based t-sne for improved visualization of single-cell rna-seq data. *Nature meth-
580 ods*, 16(3):243–245, 2019.
- 581
582 Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and
583 projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
- 584
585 Alexander Modell, Ian Gallagher, Emma Ceccherini, Nick Whiteley, and Patrick Rubin-Delanchy.
586 Intensity profile projection: A framework for continuous-time representation learning for dynamic
587 networks. *Advances in Neural Information Processing Systems*, 36, 2023.
- 588
589 Riccardo Rastelli and Marco Corneli. Continuous latent position models for instantaneous interac-
590 tions. *Network Science*, 11(4):560–588, 2023.
- 591
592 Paulo E Rauber, Alexandre X Falcao, Alexandru C Telea, et al. Visualizing time-dependent data
593 using dynamic t-sne. 2016.
- 594
595 Peter H Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*,
31(1):1–10, 1966.
- 596
597 Uriel Singer, Ido Guy, and Kira Radinsky. Node embedding over temporal graphs. *arXiv preprint
598 arXiv:1903.08889*, 2019.

594 Juliette Stehlé, Nicolas Voirin, Alain Barrat, Ciro Cattuto, Lorenzo Isella, Jean-François Pinton,
595 Marco Quaggiotto, Wouter Van den Broeck, Corinne Régis, Bruno Lina, et al. High-resolution
596 measurements of face-to-face contact patterns in a primary school. *PLoS one*, 6(8):e23176, 2011.
597

598 Terrence Szymanski. Temporal word analogies: Identifying lexical replacement with diachronic
599 word embeddings. In *Barzilay, R., Kan MY.(eds.) Proceedings of the 55th Annual Meeting of the*
600 *Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computa-
601 tional Linguistics, 2017.

602 Laurens Van Der Maaten. Accelerating t-sne using tree-based algorithms. *The journal of machine*
603 *learning research*, 15(1):3221–3245, 2014.
604

605 Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine*
606 *learning research*, 9(11), 2008.

607 Eduardo Faccin Vernier, Rafael Garcia, IP da Silva, João Luiz Dihl Comba, and Alexandru C Telea.
608 Quantitative evaluation of time-dependent multidimensional projection techniques. In *Computer*
609 *Graphics Forum*, volume 39, pp. 241–252. Wiley Online Library, 2020.

610 Eduardo Faccin Vernier, João Luiz Dihl Comba, and Alexandru C Telea. Guided stable dynamic
611 projections. In *Computer Graphics Forum*, volume 40, pp. 87–98. Wiley Online Library, 2021.
612

613 Kevin S Xu, Mark Klinger, and Alfred O Hero. A regularized graph layout framework for dynamic
614 network visualization. *Data Mining and Knowledge Discovery*, 27:84–116, 2013.

615 Zijun Yao, Yifan Sun, Weicong Ding, Nikhil Rao, and Hui Xiong. Dynamic word embeddings for
616 evolving semantic discovery. In *Proceedings of the eleventh acm international conference on web*
617 *search and data mining*, pp. 673–681, 2018.
618

619 Yujing Zhou, Weile Liu, Yang Pei, Lei Wang, Daren Zha, and Tianshu Fu. Dynamic network
620 embedding by semantic evolution. In *2019 International Joint Conference on Neural Networks*
621 *(IJCNN)*, pp. 1–8. IEEE, 2019.
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647