MLVis 2025: Machine Learning Methods in Visualisation for Big Data (2025)
D. Archambault, I. Nabney and J. Peltonen (Editors)

Neighbour Embeddings: Beyond Visualisation

Pierre Lambert[†1] , Edouard Couplet[1] , Michel Verleysen[1] , and John Aldo Lee[1,2]

[1]Université catholique de Louvain - ICTEAM/ELEN, Belgium
[2] Université catholique de Louvain - IREC/MIRO, Belgium

Abstract

Machine learning (ML) has brought powerful tools to the visualisation community, particularly through neighbour embeddings (NE). This family of algorithms enables the intuitive visualisation of high dimensional datasets, by representing these in 2*- or* 3*-dimensional spaces. This paper argues that as NE algorithms have progressed and diversified within the visualisation domain, they have matured into powerful yet often simple methods whose potential remains largely underutilised in broader machine learning contexts. This argument is illustrated by showing through two use cases, clustering and data preprocessing before a supervised task, how NE can contribute meaningfully with little additional algorithmic effort.*

1. Dimensionality reduction and neighbour embedding

Specialized by evolution to fit in a three-dimensional environment, the human brain excels at extracting and interpreting visual information. This helps for survival, but less so in data analytics, where high-dimensional (HD) data are common. To circumvent the issue and attempt to get the gist of HD data, methods of dimensionality reduction (DR) for visualisation were developed by the machine learning (ML) community.

Historically, one of the first DR algorithms was PCA [F.R01], which is a linear projection of the data such that the axes in the low dimensional (LD) space best capture the variance in the data. The method is a staple in ML, commonly used both for algorithmic processing (with the LD dimensionality generally higher than 2 or 3) and for data visualisation (LD dimensionality below 3). The linear nature of PCA limits the amount of information that can be captured on each axes, severely hampering its usefulness when used for visualisation. To counter this limitation, the family of neighbour embeddings (NE) was introduced in 2002 by G. Hinton, with an original method called stochastic neighbor embedding (SNE) [HR02]. In this new paradigm, the data is embedded in a LD space through iterative refinements of a non-linear, non-parametric transformation of data, with the aim to preserve neighbourhoods from HD to LD. This allows for much more expressive embeddings, as witnessed in Fig. 1. The figure shows the representation of 60.10^3 images of handwritten digits in 2 dimensions, using either PCA or t-SNE [vdMH08], a method of NE significantly improving on SNE. The NE reveals clearly separated clusters in the data, corresponding to the different 10 digits despite not using any supervised signals.

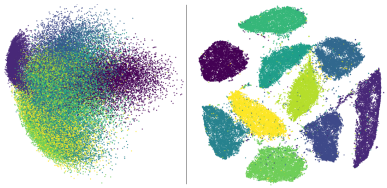

Figure 1: *PCA projection (left) and t-SNE embedding (right) of the MNIST database of handwritten digits.*

The impressive performance of NE for visualisation have propelled this family of algorithms to the centre of attention, both in applied domains [TYG*18, ADT*13, YSH*24] and in the research community. This surge in interest has led to the development of numerous methods of NE, each with unique characteristics, yet all sharing the common goal of data visualization. In short, recent advances in NE have increased their speed [MHM20, LRH*19, LCdBL24], offered more flexible ways to define neighbourhoods [KLS*20, AMK21], (perhaps inadvertently) allowed NE to embed in more than 3 dimensions [MHM20, LCdBL24], and permit new data points to be added to embeddings, either directly [LCdBL24] or in a post-hoc manner [KB19].

The purpose of this paper is to take a step back and realise that the advancements in the NE family now make them practical for direct use in a wider ML context, beyond the visualisation use case. This paper proposes two such use cases and shows that NE can be applied to these meaningfully and with minimal algorithmic modification. In section 2 we propose a novel, natural extension to NE that performs hierarchical clustering. In section 3, we show that preprocessing data using NE before a supervised task can produce embeddings which facilitate generalisation and greatly surpass other unsupervised methods, as was first hinted when parametric approximation was introduced to the community [vdM09].

† PL is a FRIA grantee of the Fonds de la Recherche Scientifique - FNRS. JAL is a Research Director with the F.R.S.-FNRS.

DOI: 10.2312/mlvis.20251156

In particular, the preprocessing application beats the state of the art one shot learning performance on ImageNet when starting from a latent representation of the images drawn from a vision-language transformer trained in a self-supervised manner.

2. Harnessing the inductive bias of NE towards clustering

Neighbour embeddings produce embeddings such that, for each point, neighbourhoods in the LD space match the neighbourhoods in the original HD space, using Euclidean distance in LD and a user-given distance metric in HD. The trick introduced by SNE allowing the formulation of this objective in derivable terms is to model neighbourhoods through a smooth kernel. Typically, the kernel is Gaussian in HD, and has heavier tails in LD. This purposeful mismatch in the HD and LD kernels is what essentially distinguishes t-SNE, as well as other recent variants of NE, from genuine SNE. Heavier tails in the LD kernel than in HD tends to shorten short-range distances, while they stretch mid- to long-range distances. This induced distance transformation, primarily meant to address issues of norm concentration [FWV07] and embedding space crowding [vdMH08], has the often observed after effect of also tightening clusters and magnifying gaps between them the LD embedding space. Broader gaps help data visualisation and ease interpretation, hypothesis formation or validation, as well as other similar exploratory tasks. Recent works [AMK21, KLS*20] have studied the effect of varying the kurtosis of the LD kernel and concluded that, the heavier the tail, the more easily the manifold will be torn in less dense regions, resulting in more clusters in the LD space. This tendency can be witnessed in Fig. 2, where embeddings of the MNIST digits datasets have been produced with increasing LD tail weights, from left to right. Looking at this figure,

Figure 2: *Embeddings of the MNIST dataset of handwritten digits with increasing LD kernel tails, from left to right.*

two observations stand out at a glance. First, the visually identifiable clusters are clearly separated in the embedding by broad strips of empty space. Second, the number of clusters in the LD space seems to relate monotonically to the weight of the LD kernel tails. From the first observation, one can intuit that a clustering algorithm running in an embedding produced by a NE algorithm would likely be less affected by the curse of dimensionality than in the original HD space, where the norms and distances tend to concentrate [FWV07]. This lead to works using NE to facilitate clustering in genomics [LSH15]. The second observation leads to the hypothesis that, if data has an underlying hierarchical structure, sweeping the kernel tail weight hyper-parameter from light to heavy tails could progressively reveal different levels of that hierarchy, with light tails showing large-scale groupings of points and heavier tails uncovering finer-grained relationships.

From these observations, one can devise a hierarchical clustering procedure. Considering a data embedding starting with light LD kernel tails, which progressively evolves as the LD kernel tails grow heavier, as if transitioning from left to right in Fig. 2. Let us take snapshots of the changing embedding at given intervals, the saved embeddings can be written $\mathcal{X}^\ell$, the level ℓ indexes the advancement in growing kernel tail weight. On each embedding $\mathcal{X}^\ell$, a clustering algorithm is performed to extract all the clusters $C^{(\ell)}$. Building on the observation that the number of clusters tends to increase when the LD tails get heavier, one can now build a graph capturing the fragmentation of large clusters into smaller ones by considering each cluster as a node, and building edges e_{ij} between $C_i^{(g)}$ and $C_j^{(h)}$ such that

$$e_{ij} = \alpha \,.\, \frac{|C_i^{(g)} \cap C_j^{(h)}|}{\min\left(|C_i^{(g)}|,\, |C_j^{(h)}|\right)}, \qquad \alpha = \begin{cases} 1 & \text{if } |h-g| = 1 \\ 0 & \text{else} \end{cases} .$$

Once the graph is obtained, one can utilise the results algorithmically or visualise them using a graph embedding algorithm, like in Figures 3 and 4, which use a force directed layout. For a cluster C_i^ℓ, the node displayed in the figures has a size proportional to $\sqrt{|C_i^\ell|}$, with bounds. The colour of the node is proper to the level ℓ of its corresponding cluster, transitioning from blue for light LD tails, to pink for heavy tails. A central node corresponding to the whole dataset has been artificially added, for aesthetic purposes. The NE was performed using an accelerated and flexible method introduced in [LCdBL24]. Concerning the hyper-parameters, the NE uses a perplexity of 35, an Euclidean distance metric in HD, and an embedding dimensionality of 4. Clustering was carried out with DBSCAN [EKSX96]. The hyper-parameters of DBSCAN were tuned manually, and this task was not as tedious as it can be when working directly with HD data, likely due to the lowered dimensionality and to the tendency of NE to overemphasise the clusters by magnifying the gaps between them.

One might observe that, given the speed of the DBSCAN algorithm along with the NE algorithm (with a complexity of $\mathcal{O}(N)$ per iteration, modern GPU implementations can embed the MNIST dataset in seconds), the proposed algorithm can be reasonably fast. Moreover it may be fully automatised by using a heuristic to drive the hyper-parameter tuning of the DBSCAN algorithm, for instance by requiring the number of clusters on level ℓ to be larger than the number of clusters found on level $\ell - 1$, ℓ increasing with the kernel tails.

Figure 3 shows a global overview of MNIST by looking at 3 levels of LD kernel tail weights, plus the central root node of the tree. The representation concurs with the 101 basics of graphology. For instance, the visually similar digits 5, 3, and 8 distinguish themselves from the rest on the upper levels of the hierarchy, before speciating into their own sub-graphs. The digits 1 with a short dash on top are closely related to poorly-written 9, and the digits 1 with a wineglass foot are considered as a mixture between 2 and 1, which makes subjective sense. Figure 4 gathers a few zoom-ins from another graph embedding with an additional level of depth, unveiling in a structured manner the myriad ways digits can be portrayed. The next section proposes another application of NE to ML: reducing the dimensionality of the data for downstream automatic processing.

3. Neighbour embedding as a pre-processing step

When using data for an automated ML task, it is common to reduce data dimensionality first, generally using PCA or sometimes autoencoders, in order to get a tractable number of meaningful features. Often-times, a good initial dimensionality reduction step can also

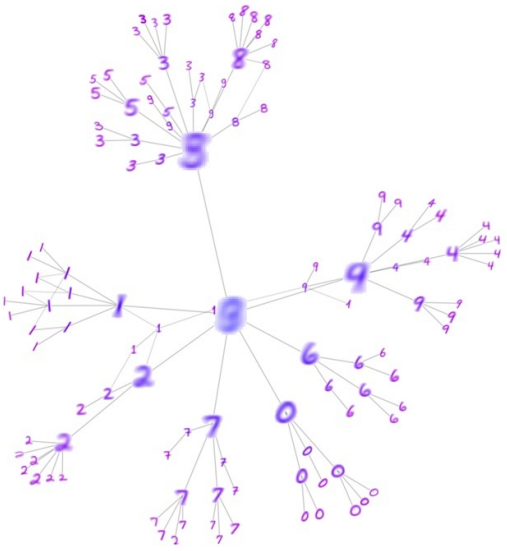

Figure 3: *Hierarchical representation of the MNIST data set of handwritten digits, using the introduced methodology of clustering.*

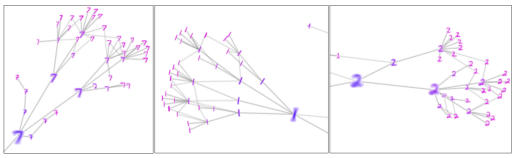

Figure 4: *Close up views of the computed graph, with one more level of depth.*

de-noise the data, improving performance in downstream tasks. A parametric approximation of t-SNE introduced in [vdM09] hints that NE can generate representations that facilitate generalisability. The capacity of modern NE to natively produce embeddings of dimensionality greater that 3 and to allow new points to be added to the embedding leads to this section, which revisits the use case of data preprocessing using non-parametric NE across 6 datasets, in Fig. 5 and Table. 1. The datasets investigated in these experiments are staples in supervised ML tasks, freely available online. For each dataset, the quality of different LD embeddings are evaluated in a one-shot learning context as well as in the more traditional 10-fold cross validation scenario. In the one shot scenario, the mean accuracy of 100 trials is reported; in each trial, one random point per label has its class revealed, and the remaining data points have their labels inferred. Every featured NE embedding is performed using an Euclidean metric in the HD space, and a arbitrarily set perplexity hyper-parameter (or the analogue `n_neighbours` in UMAP) of 35. The auto-encoder embeddings are performed using convolutional auto-encoders for MNIST and COIL-20, and standard feed-forward auto-encoders for the rest, each having one hidden layer between the latent space and the input or output layers. In the following experiments, we use XGBoost [CG16] for the 10-fold cross-validation and a 1-NN classifier for one-shot learning.

In Fig. 5, the first column shows the accuracy as a function of the embedding dimensionality in the 10-fold cross validation scenario, and the second column in the one-shot learning scenario. The dashed black line shows the performance if fitting the classification models in the original HD space. The advantage of NE is evident, as the accuracies using NE are systematically superior to those using auto-encoder or PCA, with significantly less dimensions. The gain in accuracy is impressive in one-shot learning scenarios.

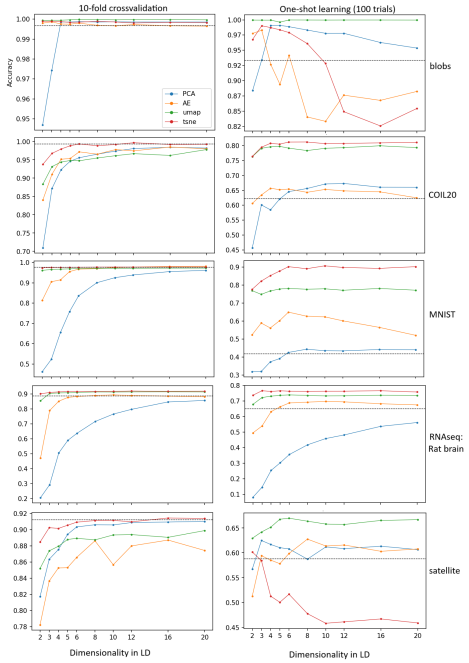

Figure 5: *Comparison of classification accuracies on embeddings produced with different DR methods.*

In addition to the 5 datasets that are reported in Fig. 5, the protocol is tested on the latent representation of the ImageNet dataset pulled from a large vision-language model called EVA [FWX*22]. In this latent space, the 1.2 million images are represented as vectors in a 1280-dimensional space. The ImageNet dataset has 1000 different classes, making high top-1 accuracies challenging to obtain. Three representations of the data are evaluated in a similar manner to the trials of Fig. 5: the bare latent representation, a PCA projection keeping the first 192 principal components with explained variance of 71.1%, and a t-SNE embedding with 32 dimensions in the latent space. These dimensionalities are chosen arbitrarily, and t-SNE ran from the 192 dimensional space of PCA rather than the initial EVA representation. Neighbour embedding was carried out using increased attractive forces and learning rate, as customary on very large data sets [KB19]. Classification performance in a 10-fold cross-validation for this dataset involved a simple multilayer perceptron with one hidden layer, for the sake of short running times. The one-shot learning evaluation involved a 1-NN classifier using Euclidean distances, as for the experiments in Fig. 5. The results of these experiments are reported in the last 4 rows of Table 1. The first 3 columns correspond to classifiers working in the EVA latent space, the PCA space, and the t-SNE space. The last column refers to a mixture of experts (MoE) [RPM*21] and shows the current state-of-

	1280, EVA	192, PCA	32, NE	MoE
zero-shot (top-1)	**84.0%**	0.0%	0.0%	
one-shot (top-1)	47.3%	45.9%	**76.2%**	68.7%
one-shot (top-5)	67.3%	67.8%	**92.0%**	
crossval (top-1, train)	84.9%	83.2%	**87.2%**	
crossval (top-1, test)	87.2%	87.2%	**88.7%**	

Table 1: *Comparison of classification performance on ImageNet across various scenarios.*

the-art top-1 one-shot accuracy using all 1000 classes on ImageNet, as reported in their paper. These experiments show that the 32-dimensional t-SNE representation enables significant improvements over the state of the art despite using a simple 1-NN classifier. The one-shot learning performance and the tighter train and test accuracies spread in the cross validation scenario indicates that the 32-dimensional t-SNE representation is a stalwart embedding and that NE may yet possess untapped potential by generating data representations that facilitate generalisation. The 32-dimensional embedding is publicly available at `https://github.com/PierreLambert3/Imagenet-tSNE-embedding`.

The elephant in the room concerns the 'zero-shot' accuracies in Table 1. Vision-language transformers consist of a text encoder and an image encoder, sharing the same latent space. They are generally trained using a contrastive loss as illustrated in Fig. 6. The shared latent space allows for so-called 'zero-shot' scenarios, where representations of inexistent images are generated by prompts like *'an image of __'* and used as artificial points with a known label, from which classification of true images can be initiated. The first row of Table 1 shows the zero-shot performance on ImageNet, using the same protocol as for the one-shot case, with the exception that the PCA projection and t-SNE embedding were performed on a dataset concatenating the representations of the actual ImageNet data points, which constitute the test set, and 1000 prototypes, one for each class label, which are generated by the text encoder. The same prototypes as in [ZA24] are used. The impressive zero-shot performance in the raw EVA space (84%) aligns with other works, while the abysmal accuracies of PCA and t-SNE in that first row call for further inquiry. A t-SNE embedding of the dataset is shown in the right panel of Fig. 6, the embedding hints that the text encoder generated points laying outside of the manifold of encoded natural images. This is confirmed by computing 'inter-label, intra-modularity' Euclidean distances in the EVA space between generated representations and comparing them to "intra-label, inter-modularity" distances between representations of true images and their generated prototypes. These distances average to 0.9 and 1.3, respectively. In other words, if ImageNet featured ducks and witches, the representations generated for the prompts 'An image of a witch' and 'An image of a duck' would lie closer to each other in the EVA space than the representations of real duck images are to their own prototype (i.e., the data point generated with 'An image of a duck'). From these observations, one can conclude that the text and image encoders have accorded themselves to produce representations featuring meaningful Euclidean distances in the context of classification using generated prototypes —this explains the high zero-shot accuracy— but the encoders have not learned to align manifolds in the latent space. In an artificial setting where prototypes for each label are generated by taking the centroid of each class in the vision encoder space (red points in 6), the top-1 accuracy in the EVA space rises from 84% for the zero-shot scenario to 85.8%. This shows that a 'better' alignment between the vision and text encoders could lead to enhanced zero-shot performances. In this example of zero-shot learning, the train set and the test set lie in separated manifolds, which is uncommon in data encountered in ML, the proposed method of pre-processing data with t-SNE to enhance generalisability fails in such cases.

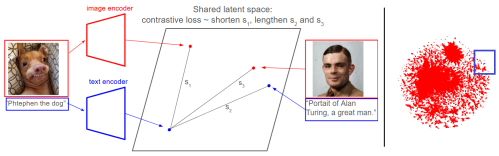

Figure 6: *Schematic explanation of vision-language models (left), and, (right) t-SNE embedding of the dataset built in the zero-shot case (red: representations of ImageNet, blue: prototypes generated from text).*

Figure 7 shows the power of the inductive bias of NE towards clustering, without the constraint of embedding in only two dimensions for visualisation. To illustrate that effect visually without further bias or complicated nonlinear distortions, 2D PCA projections are shown, starting from either the raw EVA 1280-dimensional features directly down to two dimensions or the 32D t-SNE embedding of the EVA features. The clusters clearly drift away like galaxies in an expanding universe. Notice the typical artifact of PCA projection, quite common too in spectral clustering [VL07,LS19,BN03], where only a few clusters are visually separated in the two available dimensions, appearing like long spikes, while the others collapse towards the centre of the linear projection.

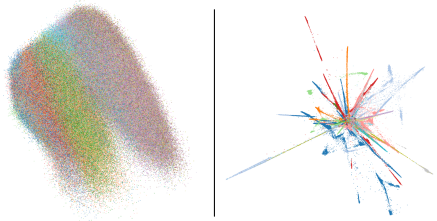

Figure 7: *Two-dimensional PCA projections of the 1280-dimensional representation of ImageNet by EVA. Left: raw EVA features (1280 (EVA) → 2 (PCA vis.)); right: EVA features after embedding in 32D with t-SNE (1280 (EVA) → 192 (PCA init.) → 32 (t-SNE) → 2 (PCA vis.)) .*

4. Conclusion

Experiments show that NE in new use cases, namely, clustering and data pre processing, can lead to meaningful representations. In particular, NE with dimensionality higher than 3 can lead to representations that generalise better, yielding ground-breaking results in scenarios of one-shot learning with simple classifiers. An exciting perspective would be to take inspirations from the capacity of NE to represent HD data robustly to devise new loss functions or regularisation terms in deep learning approaches.

References

[ADT*13] AMIR E.-A., DAVIS K., TADMOR M., SIMONDS E., LEVINE J., BENDALL S., SHENFELD D., KRISHNASWAMY S., NOLAN G., PE'ER D.: Visne enables visualization of high dimensional single-cell data and reveals phenotypic heterogeneity of leukemia. *Nature biotechnology 31* (05 2013). doi:10.1038/nbt.2594. 1

[AMK21] ABE M., MIYAO J., KURITA T.: q-SNE: Visualizing Data using q-Gaussian Distributed Stochastic Neighbor Embedding . In *2020 25th International Conference on Pattern Recognition (ICPR)* (Los Alamitos, CA, USA, Jan. 2021), IEEE Computer Society, pp. 1051–1058. URL: https://doi.ieeecomputersociety.org/10.1109/ICPR48806.2021.9412900, doi:10.1109/ICPR48806.2021.9412900. 1, 2

[BN03] BELKIN M., NIYOGI P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation 15*, 6 (2003), 1373–1396. doi:10.1162/089976603321780317. 4

[CG16] CHEN T., GUESTRIN C.: Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Aug. 2016), KDD '16, ACM, p. 785–794. URL: http://dx.doi.org/10.1145/2939672.2939785, doi:10.1145/2939672.2939785. 3

[EKSX96] ESTER M., KRIEGEL H.-P., SANDER J., XU X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (1996), KDD'96, AAAI Press, p. 226–231. 2

[F.R01] F.R.S. K. P.: Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2*, 11 (1901), 559–572. doi:10.1080/14786440109462720. 1

[FWV07] FRANCOIS D., WERTZ V., VERLEYSEN M.: The concentration of fractional distances. *IEEE Trans. Knowl. Data Eng. 19*, 7 (2007), 873–886. 2

[FWX*22] FANG Y., WANG W., XIE B., SUN Q., WU L., WANG X., HUANG T., WANG X., CAO Y.: Eva: Exploring the limits of masked visual representation learning at scale, 11 2022. doi:10.48550/arXiv.2211.07636. 3

[HR02] HINTON G. E., ROWEIS S.: Stochastic neighbor embedding. In *Advances in Neural Information Processing Systems* (2002), Becker S., Thrun S., Obermayer K., (Eds.), vol. 15, MIT Press. URL: https://proceedings.neurips.cc/paper_files/paper/2002/file/6150ccc6069bea6b5716254057a194ef-Paper.pdf. 1

[KB19] KOBAK D., BERENS P.: The art of using t-sne for single-cell transcriptomics. *Nat Commun 10, 5416* (2019). 1, 3

[KLS*20] KOBAK D., LINDERMAN G., STEINERBERGER S., KLUGER Y., BERENS P.: Heavy-tailed kernels reveal a finer cluster structure in t-sne visualisations. In *Machine Learning and Knowledge Discovery in Databases* (Cham, 2020), Brefeld U., Fromont E., Hotho A., Knobbe A., Maathuis M., Robardet C., (Eds.), Springer International Publishing, pp. 124–139. 1, 2

[LCdBL24] LAMBERT P., COUPLET E., DE BODT C., LEE J. A.: Estimated neighbour sets and smoothed sampled global interactions are sufficient for a fast approximate tsne. In *32nd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, ESANN 2024, Bruges, Belgium, October 9-11, 2024* (2024). URL: https://doi.org/10.14428/esann/2024.ES2024-203, doi:10.14428/ESANN/2024.ES2024-203. 1, 2

[LRH*19] LINDERMAN G. C., RACHH M., HOSKINS J. G., STEINERBERGER S., KLUGER Y.: Fast interpolation-based t-sne for improved visualization of single-cell rna-seq data. *Nature methods 16* (2019), 243 – 245. 1

[LS19] LINDERMAN G. C., STEINERBERGER S.: Clustering with t-sne, provably. *SIAM Journal on Mathematics of Data Science 1*, 2 (2019), 313–332. URL: https://doi.org/10.1137/18M1216134, arXiv:https://doi.org/10.1137/18M1216134, doi:10.1137/18M1216134. 4

[LSH15] LUX M., SCZYRBA A., HAMMER B.: Automatic discovery of metagenomic structure. In *2015 International Joint Conference on Neural Networks (IJCNN)* (2015), pp. 1–8. doi:10.1109/IJCNN.2015.7280500. 2

[MHM20] MCINNES L., HEALY J., MELVILLE J.: Umap: Uniform manifold approximation and projection for dimension reduction, 2020. arXiv:1802.03426. 1

[RPM*21] RIQUELME C., PUIGCERVER J., MUSTAFA B., NEUMANN M., JENATTON R., PINTO A. S., KEYSERS D., HOULSBY N.: Scaling vision with sparse mixture of experts. In *Proceedings of the 35th International Conference on Neural Information Processing Systems* (Red Hook, NY, USA, 2021), NIPS '21, Curran Associates Inc. 3

[TYG*18] TASIC B., YAO Z., GRAYBUCK L. T., SMITH K. A., NGUYEN T. N., BERTAGNOLLI D., GOLDY J., GARREN E., ECONOMO M. N., VISWANATHAN S., ET AL.: Shared and distinct transcriptomic cell types across neocortical areas. *Nature 563*, 7729 (2018), 72–78. 1

[vdM09] VAN DER MAATEN L.: Learning a parametric embedding by preserving local structure. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics* (Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009), van Dyk D., Welling M., (Eds.), vol. 5 of *Proceedings of Machine Learning Research*, PMLR, pp. 384–391. URL: https://proceedings.mlr.press/v5/maaten09a.html. 1, 3

[vdMH08] VAN DER MAATEN L., HINTON G.: Visualizing data using t-sne. *Journal of Machine Learning Research 9*, 86 (2008), 2579–2605. URL: http://jmlr.org/papers/v9/vandermaaten08a.html. 1, 2

[VL07] VON LUXBURG U.: A tutorial on spectral clustering. *Statistics and computing 17*, 4 (2007), 395–416. 4

[YSH*24] YANG W., SUN C., HUSZÁR R., HAINMUELLER T., KISELEV K., BUZSÁKI G.: Selection of experience for memory by hippocampal sharp wave ripples. *Science 383*, 6690 (2024), 1478–1483. URL: https://www.science.org/doi/abs/10.1126/science.adk8261, arXiv:https://www.science.org/doi/pdf/10.1126/science.adk8261, doi:10.1126/science.adk8261. 1

[ZA24] ZANELLA M., AYED I. B.: On the test-time zero-shot generalization of vision-language models: Do we really need prompt learning?, 2024. URL: https://arxiv.org/abs/2405.02266, arXiv:2405.02266. 4