Visualizing Similarity Data with a Mixture of Maps

James Cook, Ilya Sutskever, Andriy Mnih and Geoffrey Hinton
Department of Computer Science
University of Toronto
Toronto, Ontario M5S 3G4
located at: /h/71/hinton/papers/am/am1.tex

Abstract

We show how to visualize a set of pairwise similarities between objects by using several different two-dimensional maps, each of which captures different aspects of the similarity structure. When the objects are ambiguous words, for example, different senses of a word occur in different maps, so "river" and "loan" can both be close to "bank" without being at all close to each other. Aspect maps resemble clustering because they model pair-wise similarities as a *mixture* of different types of similarity, but they also resemble local multi-dimensional scaling because they model each type of similarity by a two-dimensional map. We demonstrate our method on a toy example, a database of human word-association data, a large set of images of handwritten digits, and a set of feature vectors that represent words.

1 Introduction

Given a large set of objects and the pairwise similarities between them, it is often useful to visualize the similarity structure by arranging the objects in a two-dimensional space in such a way that similar pairs lie close together. Methods like principal components analysis (PCA) or metric multi-dimensional scaling (MDS) [2] are simple and fast, but they minimize a cost function that is far more concerned with modeling the large dissimilarities than the small ones. Consequently, they do not provide good visualizations of data that lies on a curved low-dimensional manifold in a high dimensional space because they do not reflect the distances along the manifold [8]. Local MDS [7] and some more recent methods such as local linear embedding (LLE) [6], maximum variance unfolding [9], or stochastic neighbour embedding (SNE) [3] attempt to model local distances (strong similarities) accurately in the two-dimensional visualization at the expense of modeling larger distances (small similarities) inaccurately.

The SNE objective function is difficult to optimize efficiently, but it leads to much better solutions than methods such as LLE, and because SNE is based on a probabilistic model, it suggests a new approach to producing better visualizations: Instead of using just one two-dimensional map as a model of the similarities between objects, use many different two-dimensional maps and combine them into a single model of the similarity data by treating them as a mixture model. This is not at all the same as finding, say, a four-dimensional map and then displaying two orthogonal two-dimensional projections [6]. In that case, the four-dimensional map is the *product* of the two two-dimensional maps and a projection can be very misleading because it can put points that are far apart in 4-D close together in 2-D. In a mixture of maps, being close together in *any* map means that two objects really are similar in the mixture model.

2 Stochastic Neighbor Embedding

SNE starts by converting high-dimensional distance or similarity data into a set of conditional probabilities of the form $p_{j|i}$, each of which is the probability that one object, i, would stochastically pick another object j as its neighbor if it was only allowed to pick one neighbor. These conditional probabilities can be produced in many ways. In the word association data we describe later, subjects are asked to pick an associated word, so $p_{j|i}$ is simply the fraction of the subjects who pick word j when given word i. If the data consists of the coordinates of objects in a high-dimensional Euclidean space, it can be converted into a set of conditional probabilities of the form $p_{j|i}$ for each object i by using a spherical Gaussian distribution centered at the high-dimensional position of i, $\mathbf{x}_i$, as shown in figure 1. We set $p_{i|i} = 0$, and for $j \neq i$,

$$p_{j|i} = \frac{\exp(-||\mathbf{x}_i - \mathbf{x}_j||^2/2\sigma_i^2)}{\sum_{k \neq i} \exp(-||\mathbf{x}_i - \mathbf{x}_k||^2/2\sigma_i^2)} \qquad (1)$$

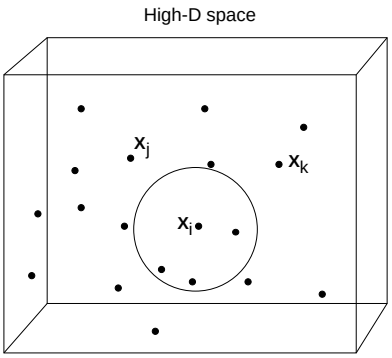

Figure 1: A spherical Gaussian distribution centered at $\mathbf{x}_i$ defines a probability density at each of the other points. When these densities are normalized, we get a probability distribution, P_i, over all of the other points that represents their similarity to i.

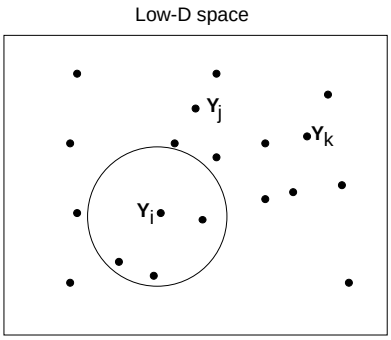

Figure 2: A circular Gaussian distribution centered at $\mathbf{y}_i$ defines a probability density at each of the other points. When these densities are normalized, we get a probability distribution over all of the other points that is our low-dimensional model, Q_i of the high-dimensional P_i.

The same equation can be used if we are only given the pairwise distances between objects, $||\mathbf{x}_i - \mathbf{x}_j||$. The variance of the Gaussian, σ_i^2, can be adjusted to vary the entropy of the distribution P_i which has $p_{j|i}$ as a typical term. If σ_i^2 is very small the entropy will be close to 0 and if it is very large the entropy will be close to $\log_2(N-1)$, where N is the number of objects. We typically pick a number $M \ll N$ and adjust σ_i^2 by binary search until the entropy of P_i is within some small tolerance of $\log_2 M$.

The goal of SNE is to model the $p_{j|i}$ by using conditional probabilities, $q_{j|i}$, that are determined by the locations $\mathbf{y}_i$ of points in a low-dimensional space as shown in figure 2:

$$q_{j|i} = \frac{\exp(-||\mathbf{y}_i - \mathbf{y}_j||^2)}{\sum_{k \neq i} \exp(-||\mathbf{y}_i - \mathbf{y}_k||^2)} \tag{2}$$

For each object, i, we can associate a cost with a set of low-dimensional $\mathbf{y}$ locations by using the Kullback-Liebler divergence to measure how well the distribution Q_i models the distribution P_i

$$C = \sum_i KL(P_i||Q_i) = \sum_i \sum_{j \neq i} p_{j|i} \log \frac{p_{j|i}}{q_{j|i}} \tag{3}$$

To improve the model, we can move each $\mathbf{y}_i$ in the direction of steepest descent of C. It is shown in [3] that this gradient optimization has a very simple physical interpretation (see figure 3). $\mathbf{y}_i$ is attached to each $\mathbf{y}_j$ by a spring which exerts a force in the direction $\mathbf{y}_i - \mathbf{y}_j$. The magnitude of this force is proportional to the length of the spring, $||\mathbf{y}_i - \mathbf{y}_j||$, and it is also proportional to the spring stiffness which equals the mismatch $(p_{j|i} - q_{j|i}) + (p_{i|j} - q_{i|j})$. Steepest descent in the cost function corresponds to following the dynamics defined by these springs, but notice that the spring stiffnesses keep changing. Starting from small random $\mathbf{y}$ values, steepest descent finds a local minimum of C. Better local minima can be found by adding Gaussian noise to the $\mathbf{y}$ values after each update. Starting with a high noise level, we decay the noise fairly rapidly to find the approximate noise level at which structure starts to form in the low-dimensional map. A good indicator of the emergence of structure is that a small decrease in the noise level leads to a large decrease in the cost function. Then we repeat the process, starting the noise level just above the level at which structure emerges and anealing it much more gently. This allows finding low-dimensional maps that are significantly better minima of C.

2.1 Symmetric SNE

The version of SNE introduced by [3] is based on minimizing the divergences between conditional distributions. An alternative is to define a single joint distribution over all non-identical ordered pairs:

$$p_{ij} = \frac{\exp(-||\mathbf{x}_i - \mathbf{x}_j||^2/2\sigma^2)}{\sum_{k<l} \exp(-||\mathbf{x}_k - \mathbf{x}_l||^2/2\sigma^2)} \tag{4}$$

$$q_{ij} = \frac{\exp(-||\mathbf{y}_i - \mathbf{y}_j||^2)}{\sum_{k<l} \exp(-||\mathbf{y}_k - \mathbf{y}_l||^2)} \tag{5}$$

$$C_{sym} = KL(P||Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \tag{6}$$

This leads to simpler derivatives, but if one of the high-dimensional points, j, is far from all the others, all of the $p_{\cdot j}$ will be very small. To overcome this problem it is possible to replace Eq. 4 by $p_{ij} = 0.5(p_{j|i} + p_{i|j})$ where $p_{j|i}$ and $p_{i|j}$ are defined using Eq. 1. When j is far from all the other points, all of the $p_{j|i}$ will be very small, but the $p_{\cdot|j}$ will sum to 1. Even when p_{ij} is defined by averaging the conditional probabilities, we still get good low-dimensional maps using the derivatives given by Eqs. 5 and 6.

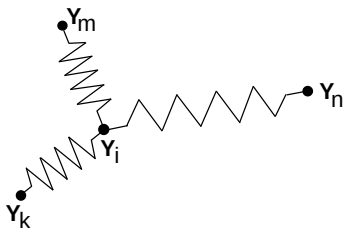

Figure 3: The gradient of the cost function in Eq. 3 with respect $\mathbf{y}_i$ has a physical interpretation as the resultant force produced by springs attaching $\mathbf{y}_i$ to each of the other points. The spring between i and j exerts a force that is proportional to its length and is also proportional to $(p_{j|i} - q_{j|i}) + (p_{i|j} - q_{i|j})$.

3 Aspect Maps

Instead of using a single two-dimensional map to define $q_{j|i}$ we can allow i and j to occur in several different two-dimensional maps. Each object, i, has a mixing proportion π_i^m in each map, m, and the mixing proportions are constrained to add to 1 for each object: $\sum_m \pi_i^m = 1$. The different maps combine to define $q_{j|i}$ as follows:

$$q_{j|i} = \frac{\sum_m \pi_i^m \pi_j^m e^{-d_{i,j}^m}}{z_i} \tag{7}$$

where

$$d_{i,j}^m = \|y_i^m - y_j^m\|^2, \qquad z_i = \sum_h \sum_m \pi_i^m \pi_h^m e^{-d_{i,h}^m}$$

Provided there is at least one map in which i is close to j and provided the versions of i and j in that map have high mixing proportions, it is possible for $q_{j|i}$ to be quite large even if i and j are far apart in all the other maps. In this respect, using a mixture model is very different from simply using a single space that has extra dimensions, because points that are far apart on one dimension cannot have a high $q_{j|i}$ no matter how close together they are on the other dimensions.

To optimize the aspect maps model, we used Carl Rasmussen's "minimize" function which is available at www.kyb.tuebingen.mpg.de/bs/people/carl/code/minimize/. The gradients are derived below.

$$\begin{aligned}
\frac{\partial C}{\partial \pi_i^m} &= -\sum_k \sum_{\ell \neq k} p_{\ell|k} \frac{\partial}{\partial \pi_i^m} [\log q_{\ell|k} z_k - \log z_k] \\
&= -\sum_k \sum_{\ell \neq k} p_{\ell|k} \left[\frac{1}{q_{\ell|k} z_k} \frac{\partial}{\partial \pi_i^m} \left(\sum_{m'} \pi_k^{m'} \pi_\ell^{m'} e^{-d_{k,\ell}^{m'}} \right) \right. \\
&\qquad \left. - \frac{1}{z_k} \frac{\partial}{\partial \pi_i^m} \left(\sum_{h \neq k} \sum_{m'} \pi_k^{m'} \pi_h^{m'} e^{-d_{k,h}^{m'}} \right) \right] \\
&= \left[-\sum_j \left(\frac{p_{j|i}}{q_{j|i} z_i} + \frac{p_{i|j}}{q_{i|j} z_j} \right) \pi_j^m e^{-d_{i,j}^m} \right] \\
&\qquad + \sum_k \sum_{\ell \neq k} \frac{p_{\ell|k}}{z_k} \frac{\partial}{\partial \pi_i^m} \left(\sum_{h \neq k} \sum_{m'} \pi_k^{m'} \pi_h^{m'} e^{-d_{k,h}^{m'}} \right) \\
&= \left[-\sum_j \left(\frac{p_{j|i}}{q_{j|i} z_i} + \frac{p_{i|j}}{q_{i|j} z_j} \right) \pi_j^m e^{-d_{i,j}^m} \right] \\
&\qquad + \sum_k \frac{1}{z_k} \frac{\partial}{\partial \pi_i^m} \left(\sum_{h \neq k} \sum_{m'} \pi_k^{m'} \pi_h^{m'} e^{-d_{k,h}^{m'}} \right) \\
&= \left[-\sum_j \left(\frac{p_{j|i}}{q_{j|i} z_i} + \frac{p_{i|j}}{q_{i|j} z_j} \right) \pi_j^m e^{-d_{i,j}^m} \right] \\
&\qquad + \sum_k \sum_{h \neq k} \frac{1}{z_k} \frac{\partial}{\partial \pi_i^m} \left(\sum_{m'} \pi_k^{m'} \pi_h^{m'} e^{-d_{k,h}^{m'}} \right) \\
&= \left[-\sum_j \left(\frac{p_{j|i}}{q_{j|i} z_i} + \frac{p_{i|j}}{q_{i|j} z_j} \right) \pi_j^m e^{-d_{i,j}^m} \right] \\
&\qquad + \sum_j \left(\frac{1}{z_i} + \frac{1}{z_j} \right) \pi_j^m e^{-d_{i,j}^m} \\
&= \sum_j \left[\frac{1}{q_{j|i} z_i} (q_{j|i} - p_{j|i}) + \frac{1}{q_{i|j} z_j} (q_{i|j} - p_{i|j}) \right] \pi_j^m e^{-d_{i,j}^m}
\end{aligned}$$

Rather than using the mixing proportions π_i^m themselves as parameters of the model, we defined paramaters w_i^m, and defined

$$\pi_i^m = \frac{e^{-w_i^m}}{\sum_{m'} e^{-w_i^{m'}}}.$$

This gives us the gradient

$$\frac{\partial C}{\partial w_i^m} = \pi_i^m \left[\left(\sum_{m'} \frac{\partial C}{\partial \pi_i^{m'}} \pi_i^{m'} \right) - \frac{\partial C}{\partial \pi_i^m} \right]$$

The distance between points i and j in map m appears as both $d_{i,j}^m$ and $d_{j,i}^m$. If $y_{i,c}^m$ denotes the cth coordinate of y_i^m, we have

$$\frac{\partial C}{\partial y_{i,c}^m} = 2 \left(\frac{\partial C}{\partial d_{i,j}^m} + \frac{\partial C}{\partial d_{j,i}^m} \right) (y_{i,c}^m - y_{j,c}^m).$$

$$
\begin{aligned}
\frac{\partial C}{\partial d_{i,j}^m} &= \sum_k \sum_{\ell \neq k} p_{\ell|k} \frac{\partial}{\partial d_{i,j}^m} (\log p_{\ell|k} - \log q_{\ell|k}) \\
&= -\sum_k \sum_{\ell \neq k} p_{\ell|k} \frac{\partial}{\partial d_{i,j}^m} \log q_{\ell|k} \\
&= -\sum_k \sum_{\ell \neq k} p_{\ell|k} \frac{\partial}{\partial d_{i,j}^m} (\log q_{\ell|k} z_k - \log z_k) \\
&= -\sum_k \sum_{\ell \neq k} p_{\ell|k} \left[\frac{1}{q_{\ell|k} z_k} \frac{\partial}{\partial d_{i,j}^m} \left(\sum_{m'} \pi_k^{m'} \pi_\ell^{m'} e^{-d_{k,\ell}^{m'}} \right) \right. \\
&\qquad \left. - \frac{1}{z_k} \frac{\partial}{\partial d_{i,j}^m} \left(\sum_{h \neq k} \sum_{m'} \pi_k^{m'} \pi_h^{m'} e^{-d_{k,h}^{m'}} \right) \right] \\
&= \frac{p_{j|i}}{q_{j|i} z_i} \pi_i^m \pi_j^m e^{-d_{i,j}^m} - \sum_\ell p_{\ell|i} \frac{1}{z_i} \pi_i^m \pi_j^m e^{-d_{i,j}^m} \\
&= \frac{p_{j|i}}{q_{j|i} z_i} \pi_i^m \pi_j^m e^{-d_{i,j}^m} - \frac{1}{z_i} \pi_i^m \pi_j^m e^{-d_{i,j}^m} \\
&= \frac{\pi_i^m \pi_j^m e^{-d_{i,j}^m}}{q_{j|i} z_i} (p_{j|i} - q_{j|i})
\end{aligned}
$$

4 Reconstructing two maps from one set of similarities

As a simple illustration of aspect maps, we constructed a toy problem in which the assumptions underlying the use of aspect maps are correct. For this toy problem, the low-dimensional space has as many dimensions as the high-dimensional space. Consider the two maps shown in figure 4. We gave each object a mixing proportion of 0.5 in each map and then used Eq. 7 to define a set of conditional probabilities $p_{j|i}$ which can be modeled perfectly by the two maps. The question is whether our optimization procedure can reconstruct both maps from one set of conditional probabilities if the objects start with random coordinates in each map. Figure 4 shows that both maps can be recovered up to reflection, translation and rotation.

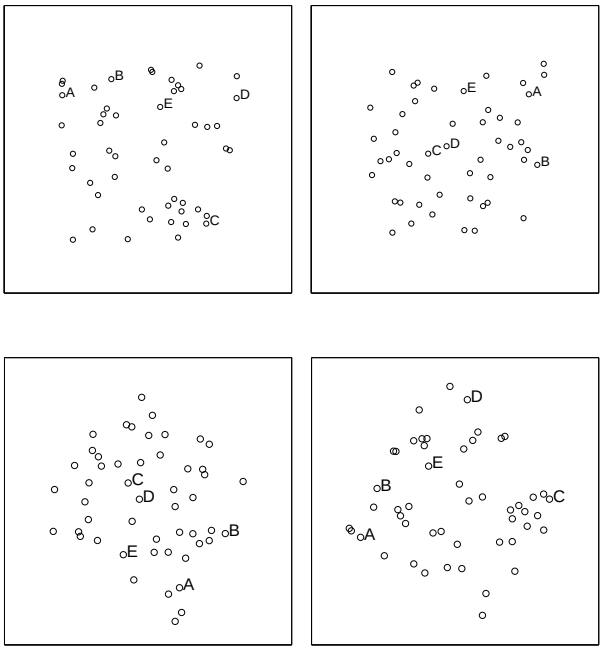

Figure 4: The two maps in the top row can be reconstructed correctly from a single set of pairwise similarities. Using a randomly chosen one-to-one mapping between points in the top two maps, the similarities are defined using Eq. 7 with all mixing proportions fixed at 0.5.

5 Modeling human word association data

The University of South Florida has made a database of human word associations available on the web. Participants were presented with a list of English words as cues, and asked to respond to each word with a word which was "meaningfully related or strongly associated" [5]. The database contains 5018 cue words, with an average of 122 responses to each. This data lends itself naturally to SNE: simply define the probability $p_{j|i}$ as the fraction of times word j was picked in response to word i.

Ambiguous words in the dataset cause a problem. For example, SNE might want to put "fire" close to the words "wood" and "job", even though "wood" and "job" should not be put close to one another. A solution is to use the aspect maps version, AMSNE, and consider the word "fire" as a mixture of two different meanings. In one map "fire" is a source of heat and should be put near "wood", and in the other "fire" is something done to employees and should be close to "job". Ambiguity is not the only reason a word might belong in two different places: as another example, "death" might be similar to words like "sad" and "cancer" but also to "destruction" and "military", even though "cancer" is not usually seen as being similar to "military".

When modelling the free association data, we found that AMSNE would put many unrelated clusters of words in the same map far apart. To make the individual maps more coherent, we added a penalty that kept each map small, thus discouraging any one map from containing several unrelated clusters. The penalty term $\frac{\lambda}{2} \sum_i \sum_m \|y_i^m\|^2$ is simply added to the cost function in Eq. 3.

We fitted the free association data with the aspect maps model using 50 maps with λ set to 0.48. In order to speed the optimization, we only used the 1000 cue words that were most often given as responses. Four of the resulting maps are shown in figures 5 and 6. In figure 5 the two different maps model the very different similarities induced by two different meanings of the word "can". In figure 6 we see two different contexts in which the word

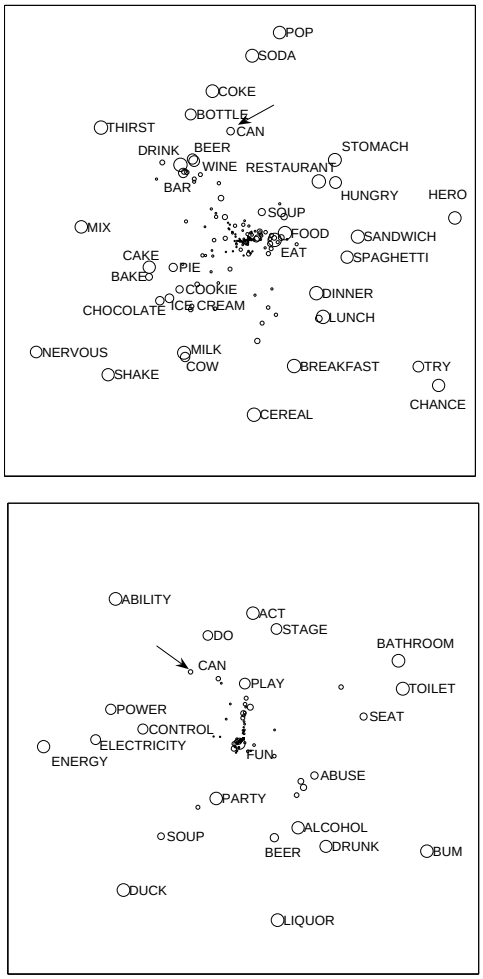

Figure 5: Two of the 50 aspect maps for the word association data. Each map models a different sense of "can". Each word is represented by a circle whose area is proportional to its mixing proportion.

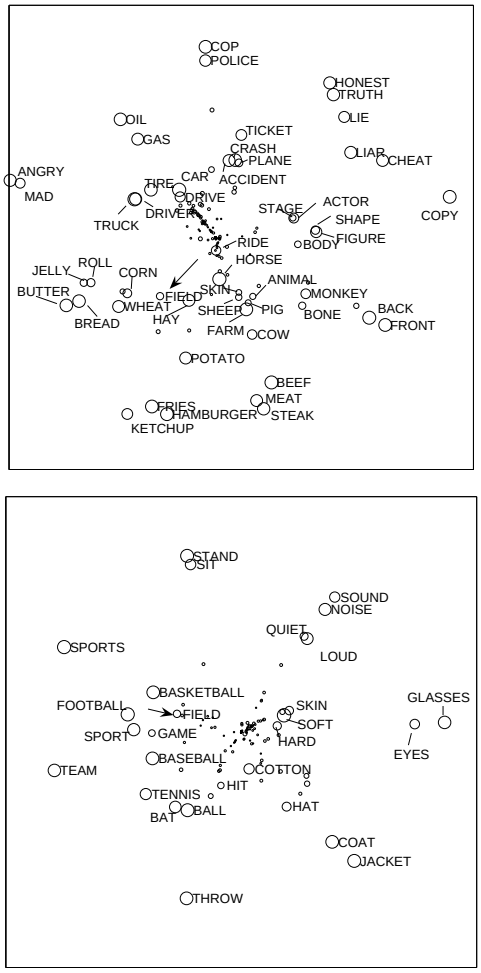

Figure 6: Two of the 50 aspect maps for the word association data. Each map models a different sense of "field".

"field" is used. Whether these should be called different meanings of the word "field" is an open question that can be answered by linguistic intuitions of lexicographers or by looking at whether two "meanings" model the observed similarity judgements better than one.

6 UNI-SNE: A degenerate version of aspect maps

On some datasets, we found that fitting two aspect maps led to solutions that seemed strange. One of the aspect maps would keep all of the objects very close together, while the other aspect map would create widely separated clusters of objects. This behaviour can be understood as a sensible way of dealing with a problem that arises when using a 2-D space to model a set of high-dimensional distances that have an intrinsic dimensionality greater than 2. In the best 2-D model of the high-dimensional distances, the objects in the middle will be crushed together too closely and the objects around the periphery will be much too far from other peripheral objects[1]. Using the physical analogy of figure 3, there will be many weak but very stretched springs between objects on opposite sides of the 2-D space and the net effect of all these springs will be to force objects in the middle together.

A "background" map in which all of the objects are very close together gives all of the $q_{j|i}$ a small positive contribution. This is sufficient to ensure that $q_{j|i}$ is at least as great as $p_{j|i}$ for objects that are significantly further apart than the average separation. When $q_{j|i} > p_{j|i}$, the very stretched springs actually repel distant objects and this causes the "foreground" map to expand, thus providing enough space to allow clusters of similar objects to be separated from each other.

If we simply constrain all of the objects in the background

[1]To flatten a hemispherical shell into a disk, for example, we need to compress the center of the hemisphere and stretch or tear its periphery.

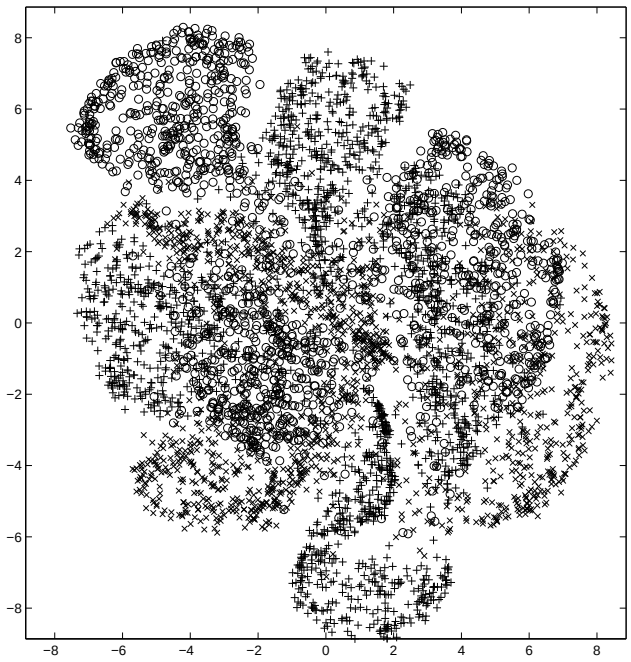

Figure 7: The result of applying the symmetric version of SNE to 5000 digit images from the MNIST dataset. The 10 digit classes are not well separated.

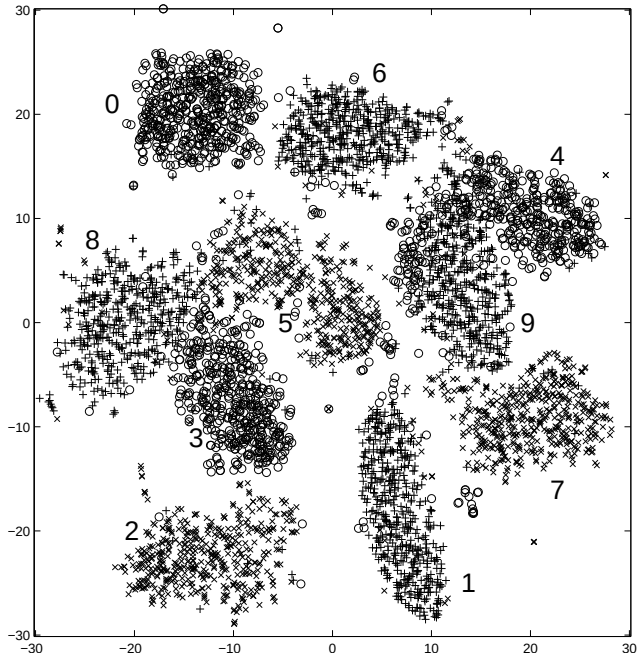

Figure 8: If 0.2 of the total probability mass is used to provide a uniform background probability distribution, the slight attraction between dissimilar objects is replaced by slight repulsion. This causes expansion and rearrangement of the map which makes the class boundaries far more apparent.

map to have identical locations and mixing proportions, we get a degenerate version of aspect maps that is equivalent to combining SNE with a uniform background model. We chose to implement this idea for the simpler, symmetric version of SNE so Eq. 5 becomes:

$$q_{ij} = \frac{(1-\lambda)\exp(-||\mathbf{y}_i - \mathbf{y}_j||^2)}{\sum_{k<l}\exp(-||\mathbf{y}_k - \mathbf{y}_l||^2)} + \frac{2\lambda}{N(N-1)} \quad (9)$$

We call this robust version "UNI-SNE" and it often gives much better visualizations than SNE. We tested UNI-SNE on the MNIST dataset of handwritten digit images. It is very difficult to embed this data into a 2-D map in such a way that very similar images are close to one another and the class structure of the data is apparent. Using the first two principal components, for example, produces a map in which the classes are hopelessly scrambled [4]. A nonlinear version of PCA [4] does much better but still fails to separate the individual classes within the clusters 4,7,9 and 3,5,8.

We first used principal components analysis on all 60,000 MNIST training images to reduce each 28×28 pixel image to a 30-dimensional vector. Then we applied the symmetric version of SNE to 5000 of these 30-dimensional vectors with an equal number from each class. To get the p_{ij} we averaged $p_{i|j}$ and $p_{j|i}$ each of which was computed using a perplexity of 30 (see [3] for details). We ran SNE with exponentially decaying jitter, stopping after 1100 parameter updates when the KL divergence between the p_{ij} and the q_{ij} was changing by less than $.0001$ per iteration. Figure 7 shows that SNE is also unable to separate the clusters 4,7,9 and 3,5,8 and it does not cleanly separate the clusters for 0, 1, 2, and 6 from the rest of the data. Starting with the solution produced by symmetric SNE, we ran UNI-SNE for a further 1500 parameter updates with no jitter but with 0.2 of the total probability mass uniformly distributed between all pairs. Figure 8 shows that this produced a dramatic improvement in revealing the true structure of the data. It also reduced the KL divergence in Eq. 6 from 2.47 to 1.48. UNI-SNE is better than any other visualization method we know of for separating the classes in this dataset, though we have not compared it with the recently developed method called "maximum variance unfolding" [9] which, like UNI-SNE, tries to push dissimilar objects far apart.

We have also tried applying UNI-SNE to a set of 100-dimensional real-valued feature vectors each of which represents one of the 500 most common words or symbols in a dataset of AP newswire stories[1]. The corpus contains 16,000,000 words and a feature vector was extracted for each of the 18,000 commonest words or symbols by fitting a model (to be described elsewhere) that tries to predict the features of the current word from the features of the two previous words. We used UNI-SNE to see whether the learning procedure was extracting sensible representations

of the words. Figure 9 shows that the feature vectors capture the strong similarities quite well.

Acknowledgments

We thank Sam Roweis for helpful discussions and Josh Tenenbaum for telling us about the free association dataset. This research was supported by NSERC, CFI and OTI. GEH is a fellow of CIAR and holds a CRC chair.

References

[1] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin. A neural probabilistic language model. *Journal of Machine Learning Research*, 3(6):1137–1155, 2003.

[2] M.A.A. Cox and T.F. Cox. *Multidimensional Scaling*. Chapman & Hall/CRC, 2001.

[3] G. Hinton and S. Roweis. Stochastic neighbor embedding. *Advances in Neural Information Processing Systems*, 15:833–840, 2003.

[4] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313:504–507, 2006.

[5] D. L. Nelson, C. L. McEvoy, and T. A. Schreiber. The university of south florida word association, rhyme, and word fragment norms. In *http://www.usf.edu/FreeAssociation/*, 1998.

[6] S.T. Roweis and L.K. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500):2323, 2000.

[7] J.W. Sammon. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, 18(5):401–409, 1969.

[8] J.B. Tenenbaum, V. Silva, and J.C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319, 2000.

[9] K.Q. Weinberger and L.K. Saul. Unsupervised Learning of Image Manifolds by Semidefinite Programming. *International Journal of Computer Vision*, 70(1):77–90, 2006.

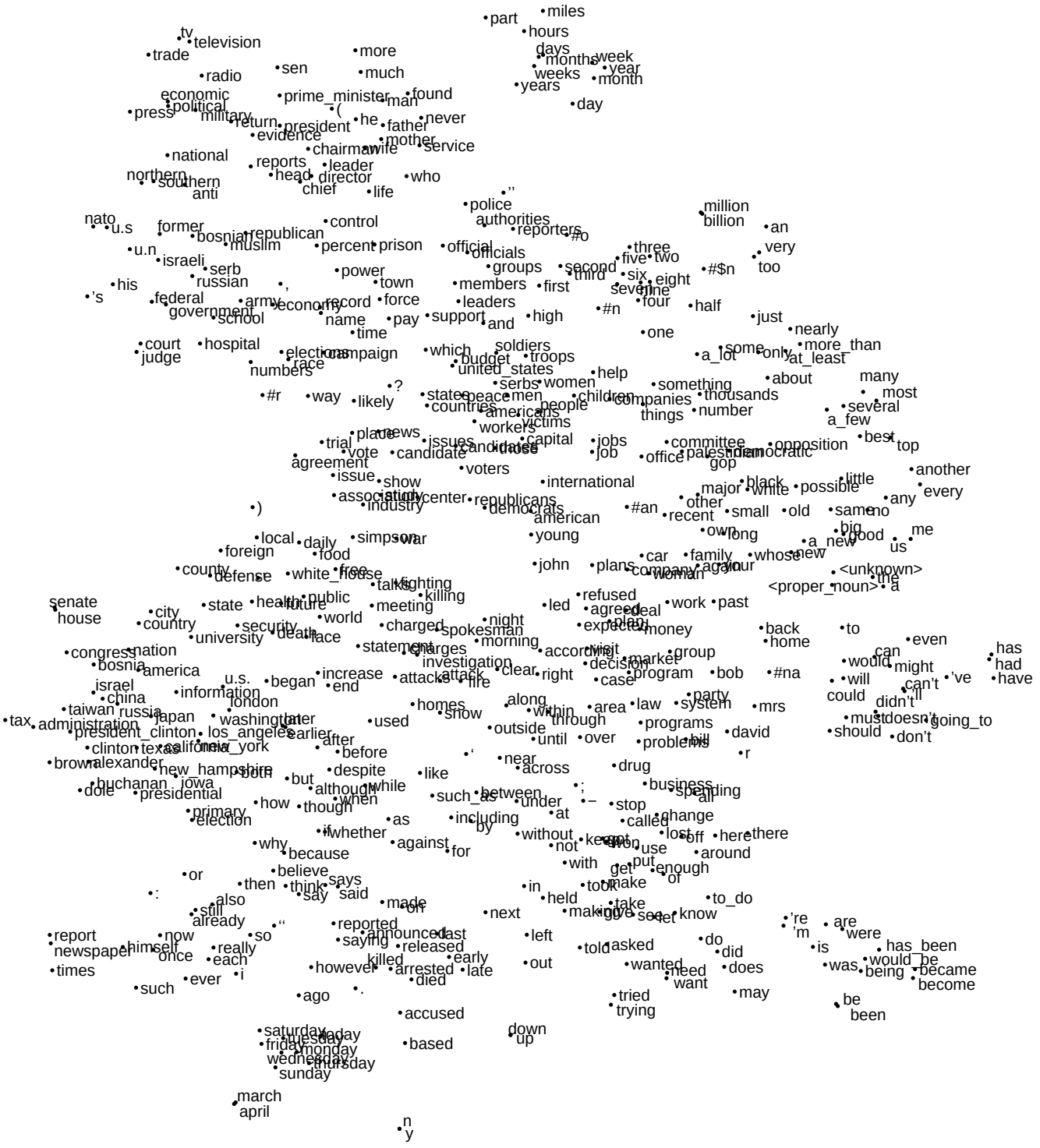

Figure 9: A map produced by applying UNI-SNE to 100-dimensional feature vectors that were learned for the 500 commonest words in the AP news dataset.