2012 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 23–26, 2012, SANTANDER, SPAIN

STOCHASTIC TRIPLET EMBEDDING

Laurens van der Maaten

Delft University of Technology
Mekelweg 4, 2628 CD Delft, The Netherlands
lvdmaaten@gmail.com

Kilian Weinberger

Washington University in Saint Louis
1 Brookings Dr., Saint Louis MO 63130
kilian@seas.wustl.edu

ABSTRACT

This paper considers the problem of learning an embedding of data based on similarity triplets of the form *"A is more similar to B than to C"*. This learning setting is of relevance to scenarios in which we wish to model human judgements on the similarity of objects. We argue that in order to obtain a truthful embedding of the underlying data, it is insufficient for the embedding to satisfy the constraints encoded by the similarity triplets. In particular, we introduce a new technique called t-Distributed Stochastic Triplet Embedding (t-STE) that collapses similar points and repels dissimilar points in the embedding — even when all triplet constraints are satisfied. Our experimental evaluation on three data sets shows that as a result, t-STE is much better than existing techniques at revealing the underlying data structure.

Index Terms— Partial order embedding, similarity triplets.

1. INTRODUCTION

The analysis of human similarity judgements is important in a range of fields, such as cognitive science, linguistics, and market research. Due to the recent advent of crowd sourcing, human similarity judgements analysis has recently also received significant attention in machine learning [1, 2, 3, 4, 5, 6]. In particular, a number of machine-learning techniques have been developed that facilitate the visual exploration of similarity judgements via embeddings.

Traditional multidimensional scaling methods [7] are often not equipped to construct embeddings based on human similarity judgements, as these methods require annotators to assign a continuous annotation to each pairwise similarity (for instance, a number on a Likert-scale from 0 to 1). This has the disadvantages (1) that different annotators use different "internal scales" and (2) that annotators may be inconsistent in their grading. Alternatively, non-metric multidimensional scaling methods may be employed, but these have the disadvantage that they require a full ranking of the objects in terms of their pairwise similarity; providing such rankings is time-consuming and error-prone. By contrast, it is much easier to gather partial similarity rankings by asking: "Is A more similar to B or to C?". Indeed, human judgements based on such *similarity triplets* are generally much more reliable [8].

In this paper we focus on learning a "truthful" embedding, i.e., an embedding in which similar inputs are close together and dissimilar inputs are far apart, entirely based on the similarity-triplets supervision. We show that it is insufficient to simply aim to satisfy the triplet constraints in the embedding through pairwise distances. In particular, we present experimental results which show that it is possible to construct qualitatively very different embeddings whilst satisfying the same percentage of the similarity triplets. We propose a novel technique for constructing embeddings based on similarity triplets, called t-Distributed Stochastic Triplet Embedding (t-STE). The main novelty of t-STE is that it collapses similar points and repels dissimilar points in the embedding whenever this does not result in additional constraint violations. Our experimental evaluations reveal that as a result, t-STE is much better than existing techniques at uncovering the underlying structure of the data (even though it does correctly model the same percentage of triplets as existing techniques).

2. PROBLEM FORMULATION

We assume we are provided with a set of inputs $\{\mathbf{z}_1, \ldots \mathbf{z}_N\} \subset \mathcal{Z}$, for which we have no representation suitable for learning, visualization, and comparison. There exists some (ground-truth) dissimilarity function $s(\mathbf{z}_i, \mathbf{z}_j)$ which quantifies the dissimilarity of any two inputs $\mathbf{z}_i, \mathbf{z}_j$. This function $s()$, however, is hidden to us. Instead, we are provided with a set of (noisy) triplets of indices:

$$\mathcal{T} = \{(i, j, \ell) |\ \mathbf{z}_i \text{ is more similar to } \mathbf{z}_j \text{ than } \mathbf{z}_\ell\}\,. \quad (1)$$

We assume that triplets $(i, j, \ell) \in \mathcal{T}$ correspond to $s()$ with some reasonable high probability, i.e., $(i, j, \ell) \in \mathcal{T}$ often implies that $s(\mathbf{z}_i, \mathbf{z}_j) < s(\mathbf{z}_i, \mathbf{z}_\ell)$. Similar to domain adaption [9], we ultimately do not evaluate the embedding on how well it captured the training signal, i.e., the triplets in $\mathcal{T}$, but instead on how-well it represents the (during training unknown) "ground-truth" $s()$.

978-1-4673-1026-0/12/$31.00

An intuitive example where such data might arise is music similarity [10], where $\mathbf{z}_i$ corresponds to the i^{th} artist in a collection of N artists. The triplets represent subjective user judgements about whether artist $\mathbf{z}_j$ is more like artist $\mathbf{z}_i$ than like $\mathbf{z}_\ell$. The function $s(\mathbf{z}_i, \mathbf{z}_j)$ could be a function that indicates whether artists $\mathbf{z}_i$ and $\mathbf{z}_j$ are in the same sub-genre. Although one can expect that most users would group artists within the same sub-genre together, which leads to triplets that agree with the ground-truth $s()$, one can also expect a significant portion of triplets to contradict the genre-based ground-truth $s()$. Imagine for example that two artists are from different genres (e.g., *pop* and *hip hop*), and therefore their s-distance is large, but some users group them together because they both passed away (e.g., *Michael Jackson* and *2Pac*). Other examples could be images of objects [1] or texture patterns [5]. Our goal is to find an embedding $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \in \mathbb{R}^r$, for some $r \ll N$, such that triplet comparisons based on Euclidean distances agree with those based on $s()$. More formally, we aim that for any $(i, j, \ell) \in \mathcal{T}$ the following relation holds with high probability:

$$\|\mathbf{x}_i, \mathbf{x}_j\|_2 < \|\mathbf{x}_i, \mathbf{x}_\ell\|_2 \iff s(\mathbf{z}_i, \mathbf{z}_j) < s(\mathbf{z}_i, \mathbf{z}_\ell). \quad (2)$$

For notational simplicity, we define the $r \times N$ design matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$ and the kernel matrix $\mathbf{K} = \mathbf{X}^\top \mathbf{X}$. Throughout this paper we use bold font to denote vectors $(\mathbf{x}_i)$, bold capital letters to denote matrices $(\mathbf{K})$ and italic font for scalars (ℓ). The i, j-entry of matrix $\mathbf{K}$ is expressed as k_{ij}.

3. EXISTING TECHNIQUES

In this section, we briefly review two recent techniques that were designed to learn data embeddings based on similarity triplets: (1) generalized non-metric multidimensional scaling and (2) crowd kernel learning.

Generalized Non-Metric Multidimensional Scaling. GNMDS aims to find a low-rank kernel matrix $\mathbf{K}$ in such a way that the pairwise distances between the embedding of the objects $\mathbf{x}_i$ in the RKHS satisfy the triplet constraints in the set $\mathcal{T}$ with a large margin [1]. GNMDS minimizes the trace-norm of the kernel $\mathbf{K} = \mathbf{X}^\top \mathbf{X}$ in order to approximately minimize its rank, which leads to a convex minimization problem. After introducing a slack variable for each constraint, the problem takes the form:

$$\min_{\mathbf{K}} \ \text{trace}(\mathbf{K}) + C \sum_{\forall (i,j,\ell) \in \mathcal{T}} \xi_{ij\ell} \text{ subject to:}$$
$$(1) \quad k_{jj} - 2k_{ij} - k_{\ell\ell} + 2k_{i\ell} \leq 1 + \xi_{ij\ell}$$
$$(2) \quad \xi_{ij\ell} \geq 0$$
$$(3) \quad \mathbf{K} \succeq 0.$$

Here, C is a constant that weighs the two competing parts in the objective: the trace regularization (functioning as an approximation of a rank constraint $\text{rank}(\mathbf{K}) \leq r$) and the importance of the triplet constraints. The optimization is performed by a type of projected gradient descent, i.e., by iteratively taking a subgradient step and projecting the resulting kernel $\mathbf{K}$ back onto the positive semidefinite cone. The embedding $\mathbf{X}$ is then obtained via an SVD of $\mathbf{K}$.

Crowd Kernel Learning. CKL [5] defines probabilities that measure how well a triplet $(i, j, \ell) \in \mathcal{T}$ is modeled:

$$p_{ij\ell} = \frac{k_{ii} + k_{jj} - 2k_{ij} + \mu}{(k_{ii} + k_{jj} - 2k_{ij}) + (k_{ii} + k_{\ell\ell} - 2k_{i\ell}) + 2\mu},$$

where μ is a small scalar value that regularizes the final solutions and prevents numerical problems. Hence, a higher probability $p_{ij\ell}$ indicates that a triplet is less well modeled. CKL learns the kernel by minimizing the empirical log-loss:

$$\min_{\mathbf{K}} \sum_{\forall (i,j,\ell) \in \mathcal{T}} \log\left(p_{ij\ell}\right) \text{ subject to:}$$
$$(1) \quad \forall i : k_{ii} = 1$$
$$(2) \quad \mathbf{K} \succeq 0.$$

The scale constraint is necessary because the objective is inherently scale-invariant. As in GNMDS, learning in CKL is performed using projected gradient descent and the embedding $\mathbf{X}$ is obtained via an SVD of the kernel $\mathbf{K}$. Although the above optimization problem is non-convex, the probabilistic interpretation of CKL has the advantage that it facilitates natural ways for it to be used in active learning setting.

Constraint gradients. Figure 1 shows the gradient that a triplet $(i, j, \ell) \in \mathcal{T}$ induces on the location of the points $\mathbf{x}_j$ (top row) and $\mathbf{x}_\ell$ (bottom row) as a function of $d(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|$ and $d(\mathbf{x}_i, \mathbf{x}_\ell) = \|\mathbf{x}_i - \mathbf{x}_\ell\|$ for various algorithms. (STE and t-STE are introduced in the following section.) Red colors (positive values) indicate that a point is moving in the direction of $\mathbf{x}_i$, whereas blue colors (negative values) indicate that a point is moving away from $\mathbf{x}_i$. The top-left region of each plot indicates the gradients when a constraint (i, j, ℓ) is satisfied, $d(\mathbf{x}_i, \mathbf{x}_j) \ll d(\mathbf{x}_i, \mathbf{x}_\ell)$. Bottom-right regions indicate the gradients when a constraint is strongly violated, $d(\mathbf{x}_i, \mathbf{x}_j) \gg d(\mathbf{x}_i, \mathbf{x}_\ell)$. The diagonal (bottom-left to top-right) represents the cases where the triplet relations become equalities, $d(\mathbf{x}_i, \mathbf{x}_j) \approx d(\mathbf{x}_i, \mathbf{x}_\ell)$.

The figure shows that, for a constraint $(i, j, \ell) \in \mathcal{T}$, in the case of GNMDS the gradients w.r.t. a point $\mathbf{x}_j$ or $\mathbf{x}_\ell$ depend linearly on the distance of that point to $\mathbf{x}_i$. These gradients suddenly drop to zero when a constraint is satisfied (with a margin of 1). This ignorance of already satisfied constraints of GNMDS is suboptimal, as it neglects the information represented by satisfied constraints in determining the underlying structure of the data. The CKL gradients depicted in Figure 1 reveal that CKL has a similar problem (although the decay of the gradients is more gradual than in GNMDS). In addition, CKL appears to be suffering from the problem that the gradient is only large whenever a constraint is strongly violated. This implies that CKL is mainly concerned with correcting

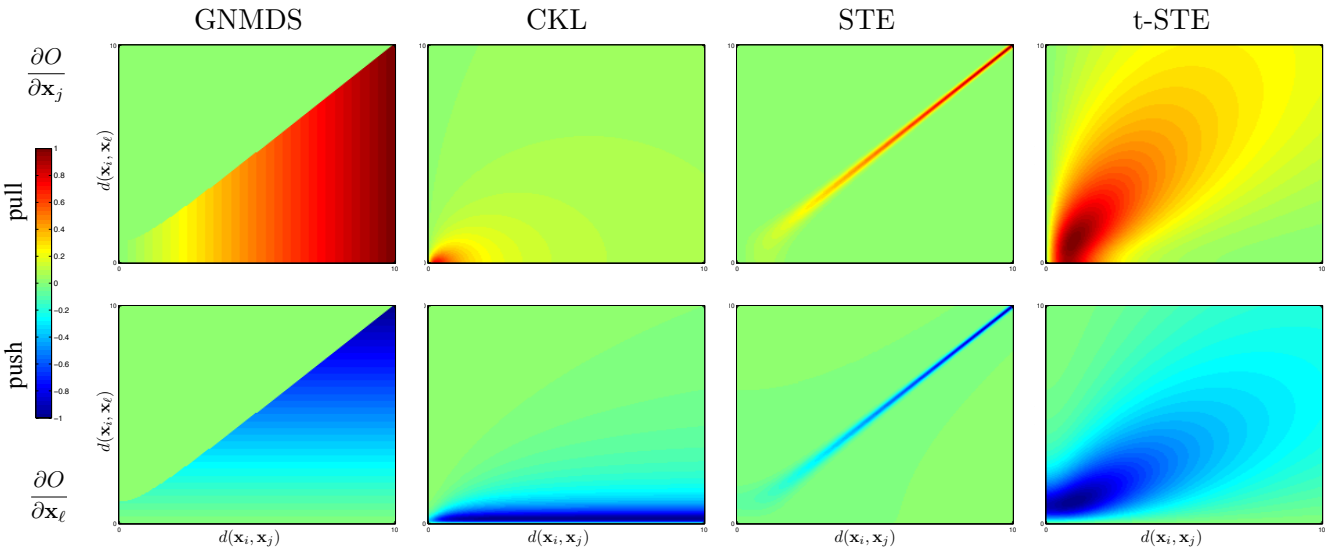

Fig. 1. Partial gradients induced by a triplet constraint for all four techniques. Figures best viewed in color.

large constraint violations in the embedding. This is suboptimal as these large constraint violations are likely to be the result of triplets that contradict the "consensus" $s()$ in the data, e.g., because they were provided by an individual with uncommon preferences. It should be noted here that when the embedding is initialized by sampling from a Gaussian with small variance, the only way a strong triplet violation may occur is if the triplet contradicts several other constraints.

4. STOCHASTIC TRIPLET EMBEDDING

To deal with the problem of CKL that it is mainly concerned with correcting constraint violations that are due to triplets that contradict the consensus, we propose a new formulation for triplet embedding that is much more *local*. In particular, our formulation (1) assigns a nearly constant penalty to large triplet violations and (2) provides a nearly constant reward for triplets that are satisfied with a large margin. Our formulation is inspired by *stochastic neighbor* approaches that have been successfully used in multidimensional scaling [11] and metric learning [12]. In particular, we define probabilities as follows:

$$p_{ij\ell} = \frac{\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2)}{\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2) + \exp(-\|\mathbf{x}_i - \mathbf{x}_\ell\|^2)}.$$

The probabilities $p_{ij\ell}$ measure the probability that the triplet (i, j, ℓ) is satisfied under a *stochastic selection rule*. Next, we aim to maximize the sum of the log-probabilities over all triplets in the training data:

$$\max_{\mathbf{X}} \sum_{\forall (i,j,\ell) \in \mathcal{T}} \log p_{ij\ell}.$$

A similar objective was also suggested in [5]. We refer to the resulting technique as Stochastic Triplet Embedding (STE).

In the formulation introduced above, the sum of triplet probabilities is maximized w.r.t. the embedding points $\mathbf{x}_i$. As an alternative, one can also maximize the objective w.r.t. the kernel matrix $\mathbf{K}$ (subject to $\mathbf{K} \succeq 0$). This leads to a convex optimization problem that can be solved via projected gradient descent. We performed experiments with such a convex variant of STE as well, using a trace-norm regularizer to minimize the rank of the kernel matrix (we obtain the final embedding via SVD).

An important difference between STE and CKL is that in STE the value of the corresponding probability rapidly becomes infinitesimal when a triplet constraint is violated. As a result, stronger violations of a constraint do not lead to significantly larger penalties, which reduces the tendency to correct triplet constraints that violate the consensus. This is illustrated by the STE gradient depicted in Figure 1: the STE gradient is nearly zero when a constraint is strongly violated or satisfied. However, it appears that the gradient decays too rapidly, making it hard for STE to fix errors made early in the optimization later on.

To address this problem, we propose to use a heavy-tailed kernel to measure local similarities between data points instead. In particular, we opt to use a Student-t kernel with α degrees of freedom by defining:

$$p_{ij\ell} = \frac{\left(1 + \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\alpha}\right)^{-\frac{\alpha+1}{2}}}{\left(1 + \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\alpha}\right)^{-\frac{\alpha+1}{2}} + \left(1 + \frac{\|\mathbf{x}_i - \mathbf{x}_\ell\|^2}{\alpha}\right)^{-\frac{\alpha+1}{2}}},$$

We refer to the resulting technique as t-Distributed STE (t-STE). The formulation of the triplet probabilities is motivated by the success of unsupervised dimensionality reduction techniques that also employ heavy-tailed similarity kernels [13,

14]. A minor disadvantage of the use of the heavy-tailed kernel is that the t-STE objective is not convex w.r.t. $\mathbf{K}$.

However, using a heavy-tailed kernel does have a major advantage over, e.g., a Gaussian kernel to compute triplet probabilities from an embedding: the resulting formulation tries to do more than simply satisfying the triplet constraints. In particular, the tails of the Student-t distribution are not *flat*, and therefore t-STE decreases the distance between points $\mathbf{x}_i$ and $\mathbf{x}_j$, even when the triplet constraint (i, j, ℓ) is already satisfied. Similarly, it increases the distance between points $\mathbf{x}_i$ and $\mathbf{x}_\ell$, even when the triplet constraint (i, j, ℓ) is satisfied. Hence, t-STE *collapses* points whenever there are no triplets keeping the points apart (i.e., when the two points represent similar objects) and it *separates* points whenever there are no triplets keeping the points together (i.e., when the two points represent dissimilar objects).

The t-STE gradient depicted in the right column of Figure 1 shows these effects: (1) the gradient w.r.t. $\mathbf{x}_j$ is attractive even when the triplet constraint is already satisfied, causing similar points to collapse, and (2) the gradient w.r.t. $\mathbf{x}_\ell$ is repulsive when the triplet constraint is already satisfied, causing dissimilar points to separate. Note that the t-STE gradient does, in contrast to the CKL gradient, decay to zero when a triplet constraint is very strongly violated. Consequently, t-STE gracefully handles the noise in $\mathcal{T}$ by not trying to satisfy constraints that contradict the consensus.

5. EXPERIMENTS

To evaluate the effectiveness of the proposed techniques, we performed experiments with STE and t-STE to compare their performance with that of GNMDS and CKL.

5.1. Experimental setup

We performed experiments on the MNIST handwritten digits data set and on a music artist similarity data set [10]. On both data sets, we assess the quality of the embeddings with two distinct metrics: (1) the percentage of held-out similarity triplets that is satisfied in the embeddings in 10-fold cross-validation experiments and (2) the leave-one-out nearest neighbor errors in the embeddings based on additional labels. These two metrics measure inherently different things. The first metric measures how well the embedding captures the training signal $\mathcal{T}$ and generalizes to new inputs from the triplet distribution. The second metric measures how well the embedding generalizes to the hidden ground-truth $s()$.

In all experiments, we considered both formulations of GNMDS, CKL, and STE that optimize w.r.t. $\mathbf{K}$ and non-convex formulations of these techniques that optimize directly w.r.t. $\mathbf{X}$ via (sub)gradient descent. The regularization parameters of the kernel variants of GNMDS and STE and the value of μ in CKL were determined by cross-validating over a wide range of parameter settings. Following [15], the number of degrees of freedom α of t-STE was set to $r - 1$. The learning rates for all techniques were fixed, and all techniques were run until convergence or until they hit a threshold of $1,000$ iterations. Code reproducing the results of our experiments is available on `http://homepage.tudelft.nl/19j49/ste`.

MNIST data set. We randomly selected a subset of $N = 1,000$ digits from the MNIST data set, and described these digits using $100,000$ triplets (i, j, ℓ), where i is picked uniformly at random, j is uniformly chosen among the 50 nearest neighbors of i, and ℓ is uniformly chosen from the set of digits that are further away from i than j (in terms of Euclidean distance between pixel values). The digit labels were *not* used in the generation of the similarity triplets.

Music artist data set. The music artist data was gathered by [10] via a web-based survey in which $1,032$ users provided $22,310$ triplets on the similarity of 426 music artists. We removed inconsistent triplets from the data using the procedure proposed by [3], leaving $9,107$ triplets on $N = 400$ artists. We also gathered genre labels for all artists using Wikipedia, distinguishing nine music genres (rock, metal, pop, dance, hip hop, jazz, country, gospel, and reggae). The genre labels were used to measure nearest neighbor errors.

5.2. Results

Below, we separately present the results of our experiments on both data sets. As a global trend across both data sets we observe that good generalization with respect to triplets does *not* translate into good nearest-neighbor classification error.

MNIST data set. The left part of Figure 2 presents the triplet generalization (measured using 10-fold cross-validation) and the leave-one-out nearest-neighbor errors of the four techniques. The results reveal that the differences between the techniques in terms of generalization to held-out triplets are relatively small in two dimensions: all techniques correctly model between 63% and 66% of the triplets, with t-STE performing slightly better than the other techniques. GNMDS and STE appear to benefit most from increasing the dimensionality of the embedding. Even though all techniques construct two-dimensional embeddings that generalize equally well when used to predict triplets, the nearest-neighbor errors in the embeddings are very different. In particular, the nearest-neighbor error of a two-dimensional t-STE embedding is 66%, whereas all other techniques produce errors of more than 80%. This suggests that embeddings which appropriately model the same amount of triplets may nonetheless have a very different local structure.

This result is supported by the two-dimensional digit maps in Figure 3, which are two-dimensional embeddings of $N = 5,000$ digits constructed based on $1,000,000$ similarity triplets[1]. All four maps in the figure have roughly the same

[1] Please note that the maps were constructed in a fully unsupervised manner, i.e., the digit labels were only used to color the points in the embedding.

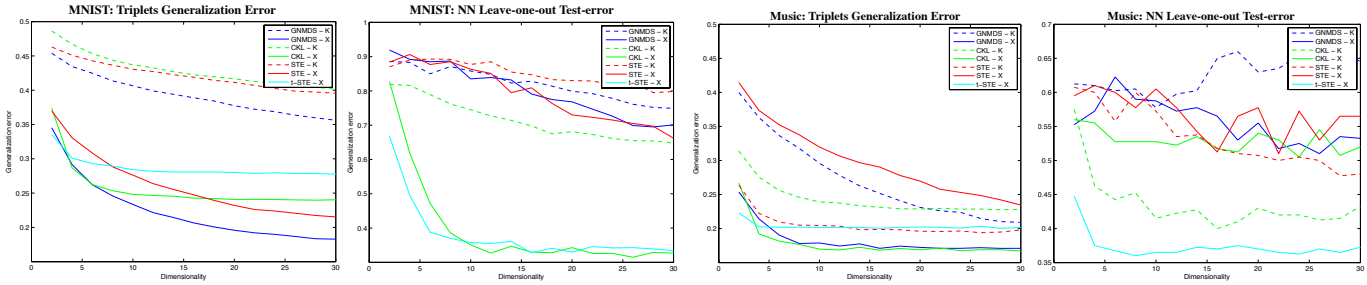

Fig. 2. Triplet generalization error of the four embedding techniques on the MNIST data set and the music artists data set *(first and third graph)*. Leave-one-out nearest neighbor errors in genre label prediction based on embeddings constructed by the four techniques on the music artists data set *(second and fourth graph)*. Figure best viewed in color.

percentage of violated triplet constraints (viz. around 20%), but the maps differ dramatically in terms of the structure they reveal. Indeed, the results nicely illustrate how t-STE differs from the other techniques in that it collapses similar points whilst repelling dissimilar points. As a result, t-STE does a much better job at separating some of the classes from the rest of the data, which leads to lower nearest-neighbor errors.

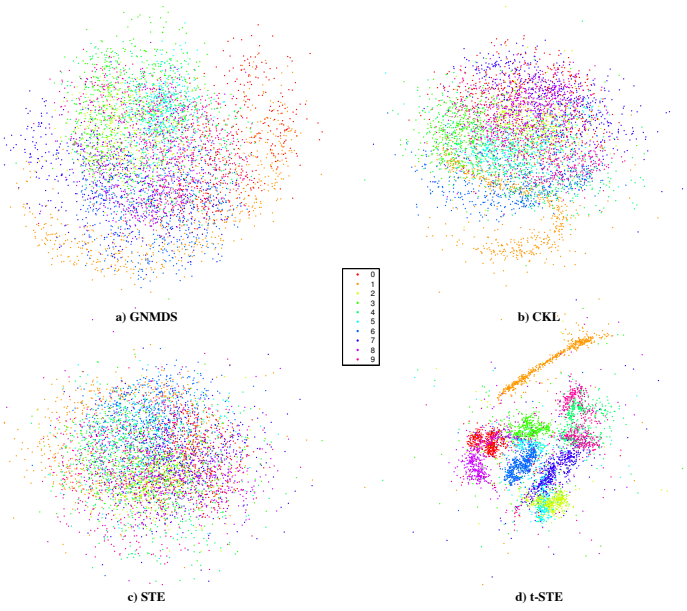

Fig. 3. Embeddings of the MNIST digit data set constructed by the four techniques. Figure best viewed in color.

Music artists data set. The right part of Figure 2 presents both quantitative metrics for the music artist data. The results are similar in that – with the exception of STE – the techniques perform roughly on par in terms of generalizing to unseen similarity triplets. The right plot shows the leave-one-out nearest neighbor errors in predicting the genre of artists based on embeddings constructed using the four techniques. Again, the results show that while all techniques perform on par in terms of generalization to unseen triplets, t-STE achieves a lower genre prediction error. The performance of t-STE is illustrated by Figure 4, which presents a two-dimensional embedding produced by t-STE on the full music artist data set (the colors of the dots correspond to the genre labels). The results shows that even in a two-dimensional embedding, t-STE is quite well capable of identifying groups of music artists who are related in terms of their genre.

6. DISCUSSION

The results presented in this paper show that there are large differences in embeddings created with different formulations of triplet embedding. We observe that it may be insufficient to only satisfy the constraints induced by the triplets. In particular, we observe that embeddings that generalize equally well to held-out triplets vary widely in terms of their nearest-neighbor errors based on the ground-truth labels.

Our proposed algorithm, t-STE, finds a triplet embedding that *collapses* inputs for which the triplet supervision provides no evidence that they are dissimilar and *separates* inputs for which there is no evidence that they are similar. The positive effect of these forces is that it allows t-STE to model the local structure of the data more effectively. Moreover, it allows for triplets that contradict the general consensus to be "overruled", which provides t-STE with better generalization to an unknown ground-truth dissimilarity. This observation is of interest to other learning problems in which the popular approaches mainly try to satisfy similarity constraints, such as non-metric multidimensional scaling and learning-to-rank.

In future work, we will study learning settings in which similarity triplets are used as side-information [16], e.g., in metric learning. This learning setting is of interest in situations in which a large amount of unlabeled data is available, and partial orderings can be obtained via crowd sourcing.

7. ACKNOWLEDGEMENTS

LvdM was supported by EU-FP7 SSPNet. KW was supported by NSF IIS-1149882 and NIH U01 1U01NS073457-01.

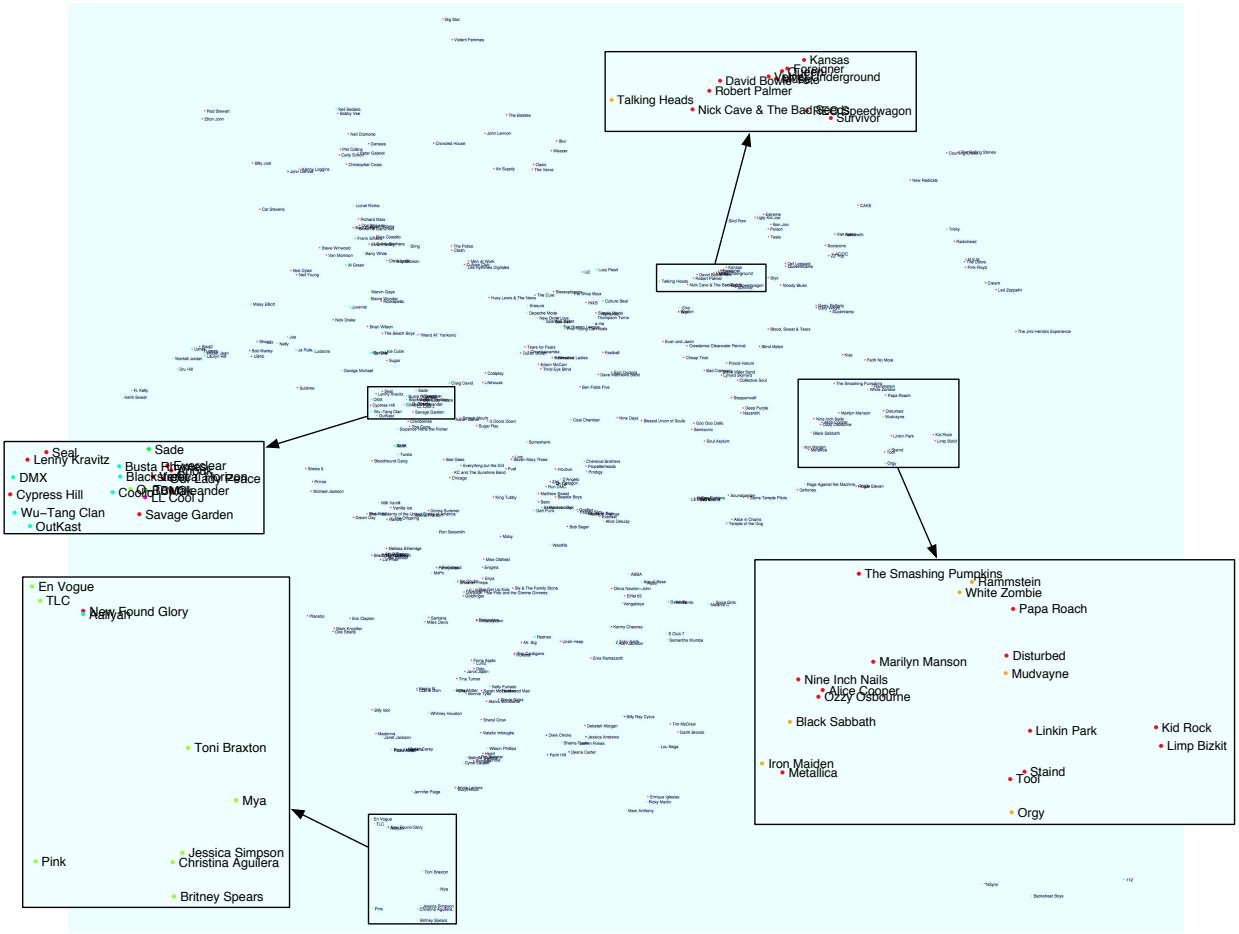

Fig. 4. Two-dimensional music artist map constructed by t-STE based on all triplets in the music artists data set. A larger version of the map is available on `http://homepage.tudelft.nl/19j49/ste`.

8. REFERENCES

[1] S. Agarwal, J. Wills, L. Cayton, G. Lanckriet, D. Kriegman, and S. Belongie, "Generalized non-metric multidimensional scaling," *JMLR W&CP 2*, pp. 11–18, 2007.

[2] K. Jamieson and R. Nowak, "Active ranking using pairwise comparisons," in *NIPS*, 2011.

[3] B. McFee and G.R.G. Lanckriet, "Learning multi-modal similarity," *JMLR*, vol. 12, no. Feb, pp. 491–523, 2011.

[4] D. Parikh and K. Grauman, "Relative attributes," in *Proc. of ICCV*, 2011, pp. 503–510.

[5] O. Tamuz, C. Liu, S.J. Belongie, O. Shamir, and A.T. Kalai, "Adaptively learning the crowd kernel," in *ICML*, 2011, pp. 673–680.

[6] L.J.P. van der Maaten and G.E. Hinton, "Visualizing non-metric similarities in multiple maps," *Machine Learning*, vol. 87, no. 1, pp. 33–35, 2012.

[7] I. Borg and P.J.F. Groenen, *Modern Multidimensional Scaling*, Springer, 2005.

[8] M. Kendall and J.D. Gibbons, *Rank Correlation Methods*, Oxford University Press, 1990.

[9] M. Chen, K.Q. Weinberger, and J. Blitzer, "Co-training for domain adaptation," in *NIPS*, 2011, pp. 2456–2464.

[10] D.P.W. Ellis, B. Whitman, A. Berenzweig, and S. Lawrence, "The quest for ground truth in musical artist similarity," in *ISMIR*, 2002.

[11] G.E. Hinton and S.T. Roweis, "Stochastic Neighbor Embedding," in *NIPS*, 2003, pp. 833–840.

[12] J. Goldberger, S. Roweis, G.E. Hinton, and R.R. Salakhutdinov, "Neighbourhood components analysis," in *NIPS*, 2005, pp. 513–520.

[13] L.J.P. van der Maaten and G.E. Hinton, "Visualizing data using t-SNE," *JMLR*, vol. 9, no. Nov, pp. 2431–2456, 2008.

[14] M.Á. Carreira-Perpiñán, "The elastic embedding algorithm for dimensionality reduction," in *ICML*, 2010, pp. 167–174.

[15] L.J.P. van der Maaten, "Learning a parametric embedding by preserving local structure," *JMLR W&CP 5*, pp. 384–391, 2009.

[16] M. Schultz and T. Joachims, "Learning a distance metric from relative comparisons," in *NIPS*, 2004.