Spaceland Embedding of Sparse Stochastic Graphs

IEEE High Performance Extreme Computing
September 25, 2019

Nikos Pitsianis\textsuperscript{1,2}  Alexandros-S. Iliopoulos\textsuperscript{2}  Dimitris Floros\textsuperscript{1}  Xiaobai Sun\textsuperscript{2}

\textsuperscript{1}Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki
\textsuperscript{2}Department of Computer Science, Duke University
Outline

1. Introduction

2. Contribution A: SG-t-SNE

3. Contribution B: SG-t-SNE-$\Pi$

4. Key references
1. Introduction
   - Graph embedding
   - Precursor work
   - Significant impact
   - Main limitations

2. Contribution A: SG-t-SNE

3. Contribution B: SG-t-SNE-Π

4. Key references
Introduction: graphs & graph embedding

Graph/network $G(V, E)$: relational data

- increasingly arise in various applications:
  - biological, social, friend networks, food webs, co-author networks, word co-occurrence networks, product co-purchase networks, . . .

Graph (vertex) embedding:

- Mapping/encoding: $V = \mathcal{X} \rightarrow \mathcal{Y} \subset \mathbb{R}^d$
  - word embedding (of a co-occurrence graph)
  - image embedding (of a nearest-similarity graph)
  - product embedding (of a co-purchase graph)
  - user embedding (of a friend network)

- to facilitate many tasks of graph data analysis
SNE: stochastic neighbor embedding algorithm

\[ X = \{x_i\}_{i=1}^n \]

\[ Y = \{y_i\}_{i=1}^n \in \mathbb{R}^d \]

\[ G(V, E_k) \quad G(V, E_k, W_k) \]

\[ k\text{NN graph} \quad \text{cast stochastic weights on } E_k \quad \text{distribution matching} \]

\[ x_i: \text{RNA sequence} \]

\[ \text{sequence embedding in } \mathbb{R}^2 \]

SNE\textsuperscript{1} pipeline illustrated with spatial embedding of
\[ n = 1,306,127 \text{ RNA sequences of E18 mouse brain cells} \]

\textsuperscript{1}Hinton and Roweis, NIPS, 2003  10x Genomics, App Note, 2017
From input vertex data $\mathcal{X} = \{ \mathbf{x}_i \}_{i=1}^n$

Find $k$NNs among $\mathbf{D} = [d^2(x_i, x_j)]_{n \times n}$

Cast $\mathbf{D}_{k\text{NN}}$ to stochastic $\mathbf{P} = [p_{ji} + p_{ij}]/2$

Vertex embedding coordinates

$\mathcal{Y} = \{ \mathbf{y}_i \}_{i=1}^n \in \mathbb{R}^d$, $d = 1, 2, 3, \ldots$

Follow t-distribution (Cauchy kernel)

$Q : q_{ij} = \frac{1}{Z} (1 + \| \mathbf{y}_i - \mathbf{y}_j \|^2)^{-1}$

Determined by the best distribution matching measured by KL divergence

$\mathcal{Y}^* = \arg \min_{\mathcal{Y}} \text{KL}(\mathbf{P} \| Q(\mathcal{Y}))$

$u$: perplexity parameter chosen by the user

$- \sum_j a_{ij} p_{ji}(\sigma_i) \log(p_{ji}(\sigma_i)) = \log(u), \quad \forall i$ (1)

---

van der Maaten and Hinton, JMLR, 2008

---

1 van der Maaten and Hinton, JMLR, 2008
t-SNE: iterative embedding process

\[ \mathcal{X} = \{x_i\}_{i=1}^n \]

\[ \mathcal{Y} = \{y_i\}_{i=1}^n \in \mathbb{R}^d \]

- **\( \mathcal{X} \)**: pixels in digit image
- **\( \mathcal{Y} \)**: digit embedding in \( \mathbb{R}^2 \)

**SNE**\(^1\) pipeline illustrated with spatial embedding of \( n = 60,000 \) handwritten digits (MNIST dataset)

---

Significant impacts

With low-dim. spatial embedding in particular, the SNE/t-SNE algorithm family has enabled

- visual inspection, identification of connections/separations
- network-based analysis for hidden connections
- hypothesis generating and scientific discoveries

Amir et al., Nat Biotechnol, 2013  
Abdelmoula et al., PNAS, 2016  
vvan Unen et al., Nat Commun, 2017
Main limitations

- Restricted to data in a metric space

- Restricted to $k$NN-based stochastic graphs

Degree $k$ and perplexity $u$ are coupled by condition $0 < u < k$ implied in (1)

Vertices of a network do not necessarily readily reside in a metric space

A typical economic phenomenon:

- low-degree nodes in majority
- hub nodes in minority

Irregular in degree distribution
Defying the parameter condition $u < \deg(i)$

Irregular degree distribution for each of three real-world networks:
Low-degree nodes (including leaf nodes) in majority; high-degree nodes in minority.
Main limitations

- Existing software programs are limited, due to slow computation speed, to
  - small graphs, or
  - 1D/2D embedding

Many networks are large;
Spacing (3D) embedding has much greater potential in preserving/encoding more structural information

(Left) kNN graph ($k = 150$) for a Möbius strip on a $256 \times 32$ lattice, with $n = 8,192$ nodes,
(Middle) 2D embedding with missed/unresolved connections,
(Right) 3D embedding with correct connections, also offering multiple or steerable views.

* van der Maaten, JMLR, 2014  https://lvdmaaten.github.io/tsne
Linderman et al., Nat Methods, 2019  https://github.com/KlugerLab/FIt-SNE
1. Introduction

2. Contribution A: SG-t-SNE
   - Admitting arbitrary stochastic graph (SG)
   - Enabled embeddings of real-world graphs

3. Contribution B: SG-t-SNE-Π

4. Key references
SG-t-SNE: stochastic graph t-SNE

\[ \mathcal{X} = \{x_i\}_{i=1}^n \]

\[ \mathcal{Y} = \{y_i\}_{i=1}^n \in \mathbb{R}^d \]

\[ G(V, E_k) \]

\[ G(V, E_k, W_k) \]

\[ G(V, E, P(\lambda)) \]

SG-SNE pipeline admitting two types of input
(top) embedding of \( n = 1,306,127 \) RNA sequences of E18 mouse brain cells
(bottom) embedding of \( n = 8,381 \) peripheral blood mononuclear cells


Pitsianis Iliopoulos Floros Sun (AUTH|Duke)
SG-t-SNE: distinctive extension & the keystone

Distinctions:

◇ Admitting arbitrary stochastic graph $P = [p_{j|i}]$
   i.e., extend the embedding to the entire family of stochastic graphs

◇ Making it feasible to exploit sparse connection pattern for
   - investigative/explorative data analysis
   - higher computation efficiency

Key: the stochastic reshaping/rescaling equations: $\forall i$

$$
\sum_j a_{ij} \phi(p_{j|i}^{\gamma_i}) = \lambda \quad \implies \quad p_{j|i}(\lambda) = \frac{a_{ij} \phi(p_{j|i}^{\gamma_i})}{\lambda},
$$

$\lambda > 0$: re-scaling parameter; $\phi \geq 0$: reshaping function, monotonically increasing

$A = [a_{ij}]$: the binary-valued adjacency matrix; Solutions $\gamma_i$ exist unconditionally

---

$^1$We used $\phi(x) = x$ for the presented embeddings
Enabled embedding of Amazon product co-purchase network

Amazon product sale network: \( n = 334,863 \) products, \( m = 1,851,744 \) edges for co-purchase connectivity, irregular degree distribution. (Left) 2D product embedding enabled by SG-t-SNE; (Right) two product clusters/subgraphs, the vertices for each are embedded closer together, with denser intra-connections.

Yang and Leskovec, K&IS, 2015
Enabled embedding of social network orkut

Social network orkut: \( n = 3,072,441 \) user nodes, \( m = 237,442,607 \) friendship links.

(Left & Middle) 3D and 2D embeddings enabled by SG-t-SNE;

(Right) Findings: There is a weak-link zone (easier to observe in 3D embedding), calibrated communities reside on one or the other side; the rich structure reflects/decodes information of geophysical regions and cultural diversities.

Yang and Leskovec, K&IS, 2015
SG-t-SNE: exploiting sparse patterns

- Vertex data: 8k peripheral blood mononuclear cells (PBMCs)
- PBMC embedding via kNN graphs by a cell similarity measure
- SG-t-SNE can use a much sparser neighbor graph

PBM cells are color coded by provided labels with the data.

Zheng et al., Nat Commun, 2017
1. Introduction

2. Contribution A: SG-t-SNE

3. Contribution B: SG-t-SNE-Π
   - Challenges in gradient updates
   - Fast calculation of sparse interactions
   - Fast calculation of dense interactions
   - Fast data translocation
   - Comparisons in performance

4. Key references
SG-t-SNE-Π: enabling spaceland (3D) embedding

\[ \mathcal{X} = \{x_i\}_{i=1}^n \]

\[ \mathcal{Y} = \{y_i\}_{i=1}^n \in \mathbb{R}^d \]

\[ \mathcal{G}(V, E_k) \]

\[ \mathcal{G}(V, E_k, W_k) \]

\[ \mathcal{G}(V, E, P(\lambda)) \]

\[ \mathcal{G}(V, E, \{\lambda\}) \]

SG-SNE-II: high-performance pipeline admitting two types of input
(top) embedding of \( n = 1,306,127 \) RNA sequences of E18 mouse brain cells
(bottom) embedding of \( n = 8,381 \) peripheral blood mononuclear cells


Embedding of Sparse Stochastic Graphs

IEEE HPEC19 | Sep 25, 2019 15 / 21
Comparison in neighborhood preservation

(Left and Middle) RNA sequence embeddings in 3D and 2D, respectively, via $k$NN graph ($k = 90$) with $n = 1,306,127$ RNA sequences of E18 mouse brain cells and 1,000 principle gene components.

(Right) Comparison in neighborhood recall shows the advantage of 3D embedding.
Iterative embedding search: computational challenges

The computation bulk is in iterative gradient updates. van der Maaten re-formulated the gradient in two interaction terms

\[
\frac{\partial}{\partial y_i} \left( \text{KL}(P||Q(Y)) \right) = \frac{4}{Z} \sum_{i \neq j} p_{ij} q_{ij} (y_i - y_j) - \frac{4}{Z} \sum_{i \neq j} q_{ij}^2 (y_i - y_j),
\]

\( \text{attractive interaction} \)

\( \text{repulsive interaction} \)

- \( PQ = [p_{ij} q_{ij}] \): kernel matrix of the attraction term, \textit{sparse} and \textit{irregular}
- \( QQ = [q_{ij} q_{ij}] \): kernel matrix of the repulsion term, \textit{full} and \textit{irregular}

with \textit{exploitable structure} by Barnes-Hut algorithm\(^1\) or by nuFFT-based factorization\(^2\)

- Both sparse and compressive interactions tend to suffer from memory latency or inadequate parallel scheduling due to irregular memory accesses

Each term in need of high-performance algorithm-software support, especially on desktop, laptop computers for individual researchers

\(^1\) van der Maaten, JMLR, 2014 \quad \(^2\) Linderman et al., Nat Methods, 2019
Accelerated gradient updates by SG-t-SNE-Π

\[
\frac{\partial (KL)}{\partial y_i} = \frac{4}{Z} \sum_{i \neq j} p_{ij} q_{ij} (y_i - y_j) - \frac{4}{Z} \sum_{i \neq j} q_{ij}^2 (y_i - y_j),
\]

attraction: \( PQ = [p_{ij} q_{ij}] \)

repulsion: \( QQ = [q_{ij} q_{ij}] \)

Fast interaction with sparse \( PQ \)
- same sparse pattern as \( P \), which is reordered (once) to a pattern of block sparse with denser blocks (BSDB)
- modified Compressed Sparse Blocks (CSB) library\(^1\)

Fast interaction with compressed \( QQ \)
- utilized an internal equi-spaced grid in two ways
- scattered data points binned into grid cells
- formulated a kernel splitting on the grid instead of augmenting the grid size by 2x in each dimension

\(^1\) Buluç et al., ASPAA, 2009  Pitsianis et al., JOSS, 2019
Multi-level data translocation in SG-t-SNE-\(\Pi\)

- By \(\Pi\) we refer to **data permutation and physical relocations** within each interaction, also in between, at every iteration step.

- The **fast data translocation problem**

  Data \(\mathcal{Y}\) available in ordering \(a(\mathcal{Y})\), to be accessed in a different ordering \(b(\mathcal{Y})\).

  Determine a data translocation scheme to carry out \(\Pi : a(\mathcal{Y}) \rightarrow b(\mathcal{Y})\) in shortest time subject to computing platform specifics.

- **Solution**: architecture-adaptive decomposition of the permutation toward
  - optimal data locality
  - maximal utilization of parallel resources
  - best payoff with data translocation overhead

\[
\Pi = \Pi_3 \cdot \Pi_2 \cdot \Pi_1
\]
Comparison in execution time for embedding of $k$NN graphs, $k \in \{30, 90\}$, with $n = 1,306,127$ nodes as single-cell RNA sequences of E18 mouse brain cells. Each embedding takes 1,000 iterations and maintains an approximation error below the same tolerance ($10^{-6}$).

1. 10x Genomics, App Note, 2017
Recap

- **SG-t-SNE**: enables embedding of arbitrary stochastic graphs
  - including kNN graphs generated by vertex data
  - embeddings of large real-world graphs reveal characteristic structures and new information

- **SG-t-SNE-II** enables fast spaceland (3D) embedding
  - preserve more neighborhood connection, structure info.
  - offer multiple vantage points
  - open source software and supplementary material at [http://t-sne-pi.cs.duke.edu](http://t-sne-pi.cs.duke.edu)

*Acknowledgments*: We thank Tiancheng Liu and anonymous reviewers for their valuable input and comments, thank Pantazis Vouzakakis and Xenofon Theodoridis for their assistance in experiments.
1. Introduction

2. Contribution A: SG-t-SNE

3. Contribution B: SG-t-SNE-Π

4. Key references


