Decomposing Large Sparse NLP Matrices

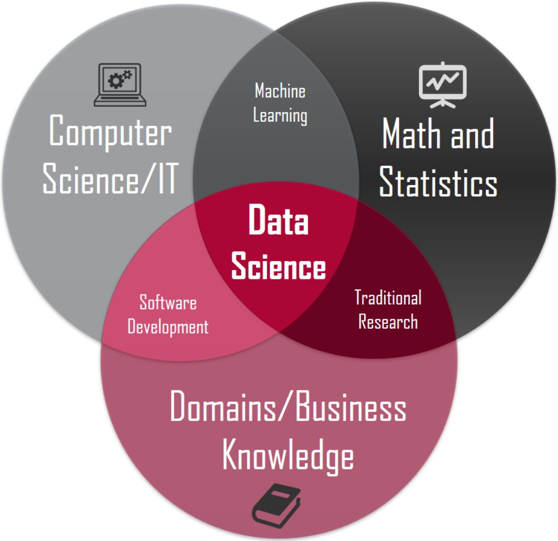
Applications in R/Python

Christopher Meaney

4th Year (Flex-Time) PhD Student, DLSPH, Division of Biostatistics, UofT
Biostatistician, Department of Family and Community Medicine, UofT
Research Student, Vector Institute
Research Student, IC/ES

January 14, 2019

Data Science with Biomedical Text Data


Computer Science/IT
Machine Learning
Math and Statistics
Data Science
Software Development
Traditional Research
Domains/Business Knowledge

Biomedical Text as Data

- Clinical Text (capture salient aspects of a clinical interaction)
  - Text that health care providers write about patients (patient EHR/EMR).
  - Text that organizations/patients write about physicians (CFPC, RateMD, others).
- Scientific Text (convey results of descriptive/analytic research)
  - Text extracted from scientific abstracts or scientific journal articles.
  - Text extracted from patents, trial registration portals, granting agencies, etc.
- Text from Social Media and the Internet
  - Text patients write about their health state (Twitter, FaceBook, Reddit, others).
  - Text patients write about health products (EBay, FDA/HC, others).
  - Text (generally) about health (Wikipedia, WebMD, NLM, CDC, WHO, others).
- Text from Genetic/Genomic Studies
  - Text encoding sequences of DNA base pairs (A,T,C,G).

Application Dataset for Today: The IC/ES Abstract Corpus

- N=3130 full text abstracts published between 1993-2018 by IC/ES scientists/staff/students.
- Data scraped from ICES/Home/Publications/Journal-Articles.

A Data and Modelling Pipeline for This Talk

- Notation: Strings ($\mathcal{S}$), Tokens ($\mathcal{T}$), Arrays ($\mathcal{X}$), Representations ($\mathcal{U}$).

$$\mathcal{S} \longrightarrow \mathcal{T} \longrightarrow \mathcal{X} \longrightarrow \mathcal{U}$$

Decomposing Large Sparse NLP Matrices

- **Objective:** Generate low-dim representation of high-dim, sparse, non-neg, NLP matrices.
- **Applications/Interpretations of Low-Dim Distributed Representation** [1]
  - Semantic, Topical, Archetypical Discovery.
  - Clustering/Browsing/Exploration.
  - Representation/Feature Learning.
  - Learn Best Latent Variable Approximation.
  - Data Compression.
  - Data Imputation. Recommendation.

Application Dataset: The IC/ES Abstract Corpus

- Simple descriptive stats on meta-data describes organization and research program.
- Low-dim representation provides topical summarization research program.
- Low-dim representation for clustering/browsing based on an archetypical abstract.

[1] Udell, M., Boyd, S., et al. (2016). Generalized Low Rank Models.

Computationally Processing Strings to Numeric Arrays

Mapping Character Sequences ($\mathcal{S}$) to Token Sequences ($\mathcal{T}$)

Low Level Computational Operations on Strings

- String splits, substrings, substitution, concatenation, etc.
- Regular expressions, pattern matching, etc.

Methods from Applied Natural Language Processing (NLP) [2]

- **Tokenization**: Convert string sequences to token sequences.
  - Token defined as an arbitrary linguistically meaningful unit of analysis.
  - Word tokenizer, sentence tokenizer, n-gram tokenizer, etc.
- **Normalization**: Stem/lemmatize, case folding, stop-words, infrequent words, etc.
- **Annotation**: Part-of-speech tagging, word shapes, etc.

Result $\mathcal{S} \rightarrow \mathcal{T}$ Pipeline: Finite dimensional set of tokens/LMUs.

- Used as an empirically defined "dictionary" of elements representing input text.

[2] Bird, S., et al. (2009). Natural Language Processing with Python.

Mapping Token Sequences ($\mathcal{T}$) to Numeric Arrays ($\mathcal{X}$)

Vector Space Models [3,4,5]

- Semantics of language captured as frequency counts of NLP matrices.
- Representation allows application math/stat models to text data.
- Representation high dimensional, sparse, non-negative integers.

Document Term Matrix (DTM): $X_{\text{DTM}} \in \mathbb{Z}_{+}^{D,V}$

- Element $\{d, v\}$ counts the number of times term/token v occurs in document d.

Term Co-Occurrence Matrix (TCM): $X_{\text{TCM}} \in \mathbb{Z}_{+}^{V,V}$

- Element $\{i, j\}$ counts number times token j occurs within some context window of token i.

Transformations: dampen/smooth impact large frequencies in matrix.

Result $\mathcal{T} \rightarrow \mathcal{X}$ Pipeline: High-dim, sparse, non-negative NLP matrices.

[3] Turney, P., Pantel, P. (2010). From Frequency to Meaning: Vector Space Models of Semantics.
[4] Lenci, A. (2018). Distributional Models of Word Meaning.
[5] Manning, C., Shutze, H. (1999). Statistical Natural Language Processing.

COMPUTATIONAL DEMONSTRATION $\mathcal{S} \rightarrow \mathcal{T} \rightarrow \mathcal{X}$ PIPELINE

Decomposing/Factorizing Large Sparse NLP Matrices

Decomposing/Factorizing Large Sparse NLP Matrices

- Processing text data yields high dimensional NLP matrices (DTM/TCM).
- Frequency counts from these matrices embody semantic structure text.
- Unsupervised learning on NLP matrices yields useful low-dim representation.

Review: Methods for (Low Rank) Reconstruction of DTM/TCM

- Multivariate Statistical Models
  - Principal Components Analysis (PCA).
  - Factor Analysis
- Low Rank Matrix Factorization/Approximation
  - Singular Value Decomposition (SVD), Latent Semantic Analysis (LSA).
  - Non-Negative Matrix Factorization (NMF).
  - Generalized Low Rank Matrix Approximations
- Bayesian Probabilistic Graphical Models
  - Latent Dirichlet Allocation (LDA) Topic Models.
  - Hierarchical Dirichlet Process (HDP) Topic Models.
  - Bayesian PCA. Bayesian Matrix Factorization.
- Neural Network Models
  - Word Vector Embedding (word2vec, GloVE). Could compose doc rep from word rep.
  - Neural Language Models. Auto-encoding documents. Neural matrix factorization.

Low Rank Matrix Approximations

Singular Value Decomposition (SVD): $X \approx U\Sigma V^T$

$$\underbrace{\begin{bmatrix} x_{1,1} & \cdots & \cdots & \cdots & x_{1,V} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{D,1} & \cdots & \cdots & \cdots & x_{D,V} \end{bmatrix}}_{D \times V \text{matrix}} \approx \underbrace{\begin{bmatrix} u_{1,1} & \cdots & u_{1,K} \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ u_{D,1} & \cdots & u_{D,K} \end{bmatrix}}_{D \times K \text{matrix}} \underbrace{\begin{bmatrix} \sigma_{1,1} & & \\ & \ddots & \\ & & \sigma_{K,K} \end{bmatrix}}_{K \times K \text{matrix}} \underbrace{\begin{bmatrix} v_{1,1} & \cdots & \cdots & \cdots & v_{1,V} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ v_{K,1} & \cdots & \cdots & \cdots & v_{K,V} \end{bmatrix}}_{K \times V \text{matrix}}$$

SVD: Best Rank-K Linear Approximation [6]

- Rank-k SVD approximation minimizes Frobenius norm $|X - U\Sigma V^T|_F^2$.
- U is D*K orthogonal matrix. Eigenvectors of XX^T.
- V is V*K orthogonal matrix. Eigenvectors of X^TX.
- Σ is K*K diagonal matrix of singular values. Square roots of eigenvalues XX^T or X^TX.

Latent Semantic Analysis: Popular model in IR/NLP/CogSci, etc. [7]

- Rows of U are K-dim representation of document.
- Rows of V^T are V-dim semantic basis for words.

[6] Eckhart, C., Young, G. (1936). The approximation of one matrix by another of lower rank.
[7] Deerwester et al. (1991). Indexing by Latent Semantic Analysis.

Non Negative Matrix Factorization (NMF): $X \approx WH$

$$\overset{D \times V \text{matrix}}{\begin{bmatrix} x_{1,1} & \cdots & \cdots & \cdots & x_{1,V} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{D,1} & \cdots & \cdots & \cdots & x_{D,V} \end{bmatrix}} \approx \overset{D \times K \text{matrix}}{\begin{bmatrix} w_{1,1} & \cdots & w_{1,K} \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ w_{D,1} & \cdots & w_{D,K} \end{bmatrix}} \overset{K \times V \text{matrix}}{\begin{bmatrix} H_{1,1} & \cdots & \cdots & \cdots & H_{1,V} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ H_{K,1} & \cdots & \cdots & \cdots & H_{K,V} \end{bmatrix}}$$

NMF: A Different Idea on the Low Rank Approximation

- Enforce non-negativity constraints on both W and H (difference from SVD). [8][9]
- Neither W nor H are orthogonal matrices (difference from SVD).
- Can constrain rows W to be non-negative, sum to one ("decomposition of parts").
- Can apply regularization (L1, L2). Yields sparse solutions. [10]

NMF: Popular model in vision/text, signal processing, time series, etc.

- Rows of W are K-dim representation of document.
- Rows of H are V-dim semantic basis for words.

---

[8] Lee, D., Seung, S. (1999). Learning Parts of Objects by Non-Negative Matrix Factorization.
[9] Lee, D., Seung, S. (2001). Algorithms for Non-Negative Matrix Factorization.
[10] Hoyer, P. (2004). Non-Negative Matrix Factorization with Sparseness Constraints.

Bayesian Probabilistic Graphical Models

Topic Models:

- Describe thematic structure of corpora via latent topics.
  - A topics is a discrete/categorical distribution over vocabulary.
- Useful for organizing, summarizing large text corpora.
- Example mixed membership model, admixture model, etc.

Mixed Membership Models (MMMs): Mixture Models for Grouped Data [11]

- Data naturally grouped ($w_{d,n}$). For $d = 1 \ldots D$ and $n = 1 \ldots N_d$.
- Each group represented as mixture model.
- Mixture components shared across groups.
- Mixture proportions vary between groups.

Generative Process for Mixed Membership Models

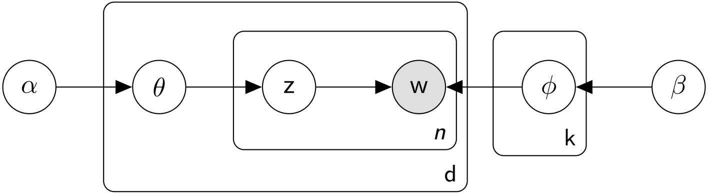
1. Draw shared components $\phi_k \sim f(\cdot|\beta)$
2. For each group $d = 1 \ldots D$:
   1. Draw proportions $\theta_d \sim \text{Dir}(\alpha)$
   2. For each data point, $w_{d,n}$, $n = 1 \ldots N_d$:
      1. Draw a mixture assignment $z_{d,n} \sim \text{Cat}(\theta_i)$
      2. Draw the data point $w_{d,n} \sim g(\cdot|\phi_{z_{d,n}})$

[11] Airoldi, E. (2014). Introduction to Mixed Membership Models and Methods.

Latent Dirichlet Allocation (LDA): a MMM for discrete/text data. [12]

1. Draw shared components $\phi_k \sim \text{Dir}_V(\cdot|\beta)$
2. For each group $d = 1...D$:
    1. Draw proportions $\theta_d \sim \text{Dir}_K(\alpha)$
    2. For each data point, $w_{d,n}$, $n = 1...N_d$:
        1. Draw a mixture assignment $z_{d,n} \sim \text{Cat}_K(\theta_i)$
        2. Draw the data point $w_{d,n} \sim \text{Cat}_V(\cdot|\phi_{z_{d,n}})$

Latent Dirichlet Allocation (LDA): a graphical model perspective.



[12] Blei, D., Ng, A., Jordan, M. (2003). Latent Dirichlet Allocation.

Latent Dirichlet Allocation (LDA): Joint/posterior Distribution

- Generative-process/graphical-model imply joint distribution
- Posterior distribution from joint distribution, by conditional probability.

$$
\begin{aligned}
P(z,\theta,\phi|w,\alpha,\beta) &= \frac{P(w,z,\theta,\phi|\alpha,\beta)}{P(w)} \\
&\propto P(w,z,\theta,\phi|\alpha,\beta) \\
&= \prod_k P(\phi_k;\beta)\prod_d P(\theta_d;\alpha)\prod_n P(z_{d,n}|\theta_d)P(w_{d,n}|\phi_{z_{d,n}}) \\
&= \prod_k \mathrm{Dir}(\phi_k;\beta)\prod_d \mathrm{Dir}(\theta_d;\alpha)\prod_n \mathrm{Cat}(z_{d,n}|\theta_d)\mathrm{Cat}(w_{d,n}|\phi_{z_{d,n}})
\end{aligned}
$$

Latent Dirichlet Allocation: Posterior Inference

- Gibbs Sampling [13]. Collapsed Gibbs Sampling [14]. Variational Inference [15].

[13] Pritchard, J. et al. (2000). Inference of Population Structure using Multilocus Genotype Data.
[14] Griffiths, T. (2002). Gibbs Sampling in the Model of Generative LDA.
[15] Blei, D. et al. (2016). Variational Inference: A Review for Statisticians.

Neural Network Models

Vector Embedding Models

- Embed/represent words using a Euclidean vector space $\mathbb{R}^p$.
- Dimension of p typically small: p=50-1000.

Goal Vector Embeddings: Distributional Hypothesis in NLP

- Semantically related words are close in vector space.
- Semantically dissimilar words are far apart in vector space.

Other interesting Properties Vector Embedding Models

- Ability to learn interesting linear latent sub-structures.
- E.g. Vector arithmetic captures analogical reasoning.
- E.g. Vector difference capture tense, pluralization, etc.
- E.g. Vector difference captures verb/adjective forms.

Algorithms for Estimating Word Vectors:

- **word2vec** [16][17]
- **GloVe** [18]

---

[16] Mikolov, T. et al. (2013a). Efficient Estimation of Word Representations in Vector Space.
[17] Mikolov, T. et al. (2013b). Distributed Representations of Words and their Compositionality.
[18] Pennington, J., et al. (2014). GloVe: Global Vectors for Word Representation.

word2vec: Comp. Graph Continuous Bag of Words (CBOW) model

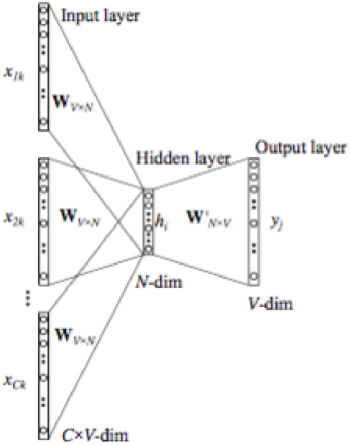


Figure Courtesy: Rong, X. (2014). word2vec Parameter Learning Explained.

word2vec: Computational Graph Skipgram Model

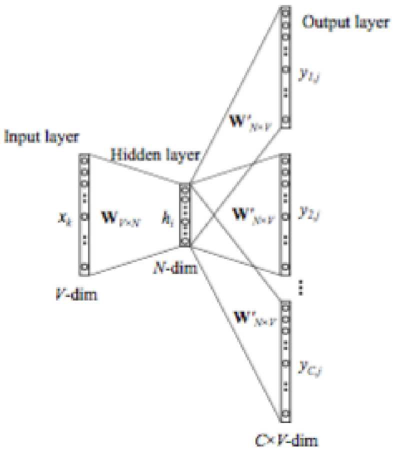


Figure Courtesy: Rong, X. (2014). word2vec Parameter Learning Explained.

GloVe: Global Vectors model

- Similar objective word2vec. Embed words $v = 1 \ldots V$ in vector space $\mathbb{R}^p$.
- Direct model elements TCM. word2vec implicit via collocation probability.
- Count vs. predict methods. [19][20]

GloVe Optimization

- Objective Function: $J = \sum_{i,j=1}^{V} f(X_{i,j})\left(w_i^T w_j + b_i + b_j - \log(X_{i,j})\right)$

$$f(x) = \begin{cases} \left(\frac{x}{x_{max}}\right)^{\alpha}, & \text{if } x < x_{max}. \\ 1, & \text{if } x \geq x_{max}. \end{cases} \tag{1}$$

- Typical choices: $\alpha = \frac{3}{4}$, $x_{\text{max}} = 100$.
- Train model via stochastic gradient descent (or variant).
  - AdaGrad popular SGD optimizer.
  - Typicaly AdaGrad learning rate $\eta = 0.05$.
  - Stochastically sample non-zero elements TCM.

[19] Baroni, M., et al. (2014). Dont Count Predict: Systematic Comparison Semantic Vectors.
[20] Levy, O., Goldberg, Y. (2014). Neural Word Embedding as Implicit Matrix Factorization.

Computational Demonstration with R/Python

Computational Demonstration with R/Python

Application Dataset: The ICES Abstract Corpus .

- N=3130 unique article titles from ICES scientists/staff/students from 1993-2018.
- Text data from scientific abstract (5.6Mb of plain text data).
- Min/max abstract length: (min=57 tokens, max=483 tokens).
- Unique words/tkens in vocabulary (cleaned): 6369.
- Total number of words/tokens in corpus (cleaned): 431,390.
- DTM dimensions and sparsity: 3130 rows, 6369 cols, 98.71% sparse.
- TCM dimensions and sparsity: 6369 rows, 6369 cols, 99.86% sparse.

Recall Goals of ICES Abstract Corpus Linguistic Analysis:

- Simple descriptive stats on meta-data describes organization and research program.
- Low-dim representation provides topical summarization research program.
- Low-dim representation for clustering/browsing based on an archetypical abstract.

R Packages and Python Modules:

- Singular Value Decomposition (SVD)
  - R Packages: `svd()`. `irlba`.
  - Python Modules: `scipy.sparse.linalg.svds`.
- Non-Negative Matrix Factorization
  - R Packages: `NNLM`. `NMF`.
  - Python Modules: `Nimfa`.
- Latent Dirichlet Allocation
  - R Packages: `text2vec`. `lda`. `topicmodels`.
  - Python Modules: `gensim`. `lda`.
- Word Vector Embeddings (word2vec, GloVe, etc.)
  - R Packages: `text2vec`.
  - Python Modules: `gensim`.