ELECTROMAGNETIC & SENSOR SYSTEMS DEPARTMENT

Wiki-World: Implicit Translation via Graph Embeddings

David J. Marchette

April 6, 2009

Indiana University

Outline

Introduction

- Data Fusion
- Wikipediæ
- Implicit Translation

Graph Embeddings

- Graph Theory Basics
- Laplacian Eigenmaps
- Random Dot Product Graphs
- Simultaneous Embeddings
- Random Graph Embeddings

Examples

- Algebraic Geometry
- English vs French

Thoughts on Glorified Word-Count Histograms

Conclusions and Future Work

Outline

Data Fusion

- Given data of different types (graphs, measurements, signals, . . .) we wish to perform inference using all the data.
- How do we combine information from two graphs? This will be the topic of this lecture.
- More generally, how do we:
  - combine information contained in a graph with other external information?
  - combine two (or more) very different types of information?
- These are fruitful areas of research.

Data Fusion Methods

$$\begin{tikzcd} D_1 \arrow[r] & \mathbb{R}^p \arrow[dr] & \\ & & \mathbb{R}^d \\ D_2 \arrow[r] & \mathbb{R}^q \arrow[ur] & \end{tikzcd}$$

Embed each independently
then combine embeddings
(simplest: $\mathbb{R}^d = \mathbb{R}^p \times \mathbb{R}^q$)

$$\begin{tikzcd} D_1 \arrow[r] & \mathbb{R}^d \arrow[dr] & \\ & & \mathbb{R}^d \\ D_2 \arrow[r] & \mathbb{R}^d \arrow[ur] & \end{tikzcd}$$

Embed into copies of a space
then combine embeddings
(eg: Procrustes)

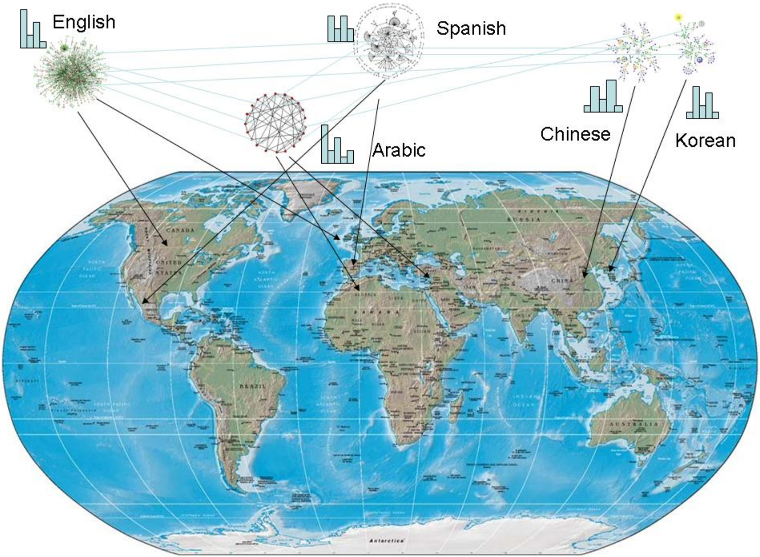
$$\begin{tikzcd} D_1 \arrow[dr] & \\ & \mathbb{R}^d \\ D_2 \arrow[ur] & \end{tikzcd}$$

Embed simultaneously
(can we do this?)

Wikipediæ

- Wikipedia is an open-source Encyclopedia that is written by a large community of users (everyone who wants to, basically).
- There are versions in over 80 languages, with various amounts of content.
- The full data for the Wikipediæ are freely available for download.
- A Wikipedia article has one or more of:
  - Title.
  - Unique ID number.
  - Text – the content of the article.
  - Internal links – links to other (related?) articles.
  - External links – links to other content on the web or elsewhere.
  - Language links – links to “the same” page in other languages.

Wiki-World


English
Spanish
Arabic
Chinese
Korean
CANADA
UNITED STATES
BRAZIL
RUSSIA
CHINA
AUSTRALIA
Antarctica

Implicit Translation

- By **implicit translation** I mean the association of documents that “mean the same” or are on the same topic.
- This is not a word-level translation, nor is it meant to imply a real translation in any normal sense.
- The idea is to associate two documents that are on the same topic:
  - The Afrikaans article titled “Sterrekunde” should be matched with the one titled “Astronomy” in English.
  - We want to do this with no word-level dictionary available.
- Note that while these articles will likely not be direct translations of each other, they are in some sense translations of the “Astronomy” topic into the two languages.
- Further, they should be associated only if they do in fact discuss the same topics.

Wiki-World as a Testbed for Implicit Translation

- Consider two Wikipedia, English and French:
  - G_E and G_F the graphs within the each Wikipedia.
  - L_E and L_F the lexicons in the two languages.
  - H_E and H_F the language models of each article (word-count histograms).
  - $A_{E \leftrightarrow F}$ the associations between the languages.
- We can treat the given associations as ground truth, and use these to determine the quality of a given method for implicit translation.
- How do we perform the implicit translation?

Implicit Translation via Embeddings

- The basic idea is as follows:
  - Embed the two sets of documents into the same space.
    - This may use the graphs, the text, or both.
    - We will focus primarily on the graphs in this talk.
  - Given any pair of documents, one from each language, determine their similarity in the space.
  - Similar pairs are associated.
- The key is the embedding: how should we embed these two graphs (two word-count histograms, . . .) into the same space?

Outline

Graphs

- A graph is a pair $G = (V, E)$, $V = [n]$, $E \subset V \times V$.
- V are called vertices, E are edges, the pairs in E are unordered and distinct.
- If the edges are ordered pairs, the graph is a directed graph (digraph), and the edges point from the first element to the second.
- Write ij for $\{i, j\} \in E$.
- A graph can be presented in many ways. We'll consider:
  - An edge list: $\{\{i_1, j_1\}, \ldots, \}$.
  - An adjacency matrix: $A = (a_{ij}), a_{ij} = I\{\{i, j\} \in E\}$.
- The degree of a vertex v is the number of edges incident to it. For directed graphs there are out-degrees and in-degrees.

Thoughts on Graph Embeddings

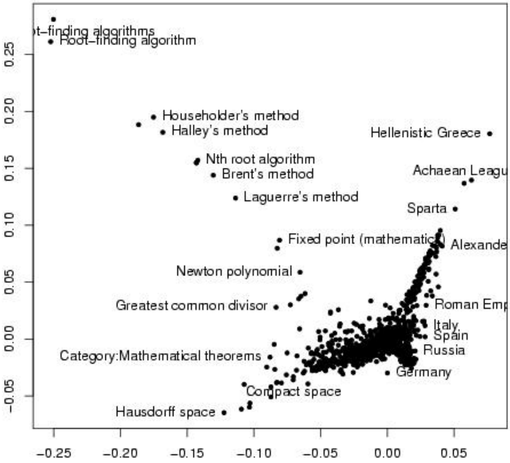
- What we want: to embed a graph into $\mathbb{R}^d$, say, so that vertices that are "close" are "close" in the space, and those that are "far" are "far".
- For implicit translation, it is not enough to embed just one graph, we want to embed two.
- Worse, we want vertices that *should be* close to end up close.
- What can we do?
  - Embed each independently and use "Procrustes" to map them together?
  - Embed each independently and use something more clever.
  - Embed them simultaneously. How?
- Can we also use the lexicons somehow?

Laplacian Eigenmaps

- Write $L = D - A$ where D is the diagonal matrix of vertex degrees, A the adjacency matrix (of a graph).
- The spectrum of L provides one way to embed the graph:
  - Compute the d eigenvectors associated with the d smallest non-zero eigenvalues.
  - Use these as the coordinates of the embedded vertices (after scaling by the eigenvalues).

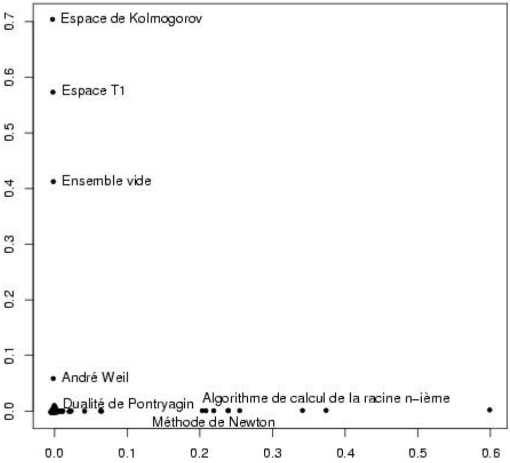
Example Laplacian Eigenmap: Good News

- Documents within two links of “Algebraic Geometry” in the English Wikipedia.
- Laplacian Eigenmap:



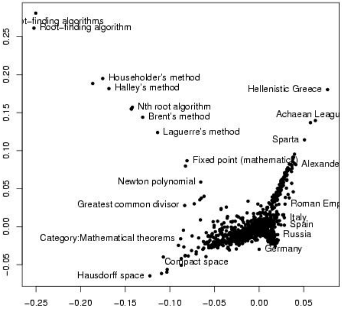
Example Laplacian Eigenmap: Bad News

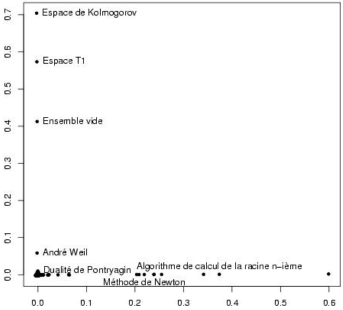
- French documents corresponding to the “Algebraic Geometry” subset from the English Wikipedia.
- Laplacian Eigenmap:



More Care is Needed

- Note that one “arm” is algorithms, one is spaces.
- This is some evidence that the French projection is related to the English, but distorted.





What Went Wrong?

- The French version is not connected. We are not being careful to account for this.
- We are not using any of the information we have about between-graph relationships.
- Small changes in the graph can produce large changes in the embedding.
- Procrustes can handle rotational ambiguity, but not distortions.

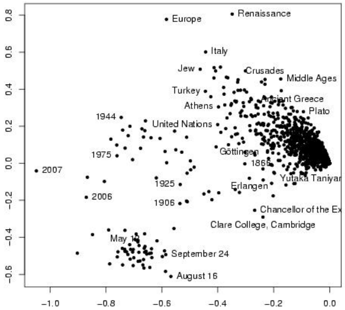
Random Dot Product Graphs

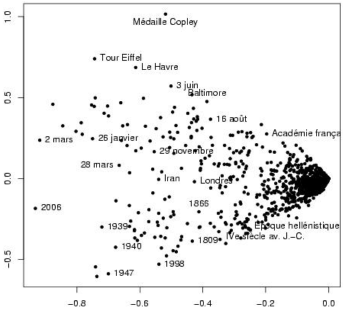
- A simple model for random graphs that is sometimes useful in social network analysis.
- Each vertex i has associated a vector x_i in a "latent space".
- $P[ij] = x_i' x_j$ with edges conditionally independent.
- For directed graphs, there are two vectors (an in- x and an out- y) and

  $$P[ij] = y_i' x_j$$

  with edges conditionally independent.
- The vectors can be estimated from a graph using spectral analysis (least squares):
  - Let $A' = A + D/(n-1)$ be the adjacency matrix augmented on the diagonal by the "probability of an edge".
  - Decompose $A' = U\Lambda$ or $A' = U\Lambda V'$ for directed graphs.
  - Scale U (V) by $\sqrt{\Lambda}$: $X = U\sqrt{\Lambda}, Y = V\sqrt{\Lambda}$.
- Embed as X or Y or (X, Y).

Better?


Europe
Renaissance
Italy
Jew
Crusades
Middle Ages
Turkey
Athens
Ancient Greece
Plato
1944
United Nations
1975
Göttingen
1866
2007
Yutaka Taniyar
1925
Erlangen
2006
1906
Chancellor of the Ex
Clare College, Cambridge
May 1
September 24
August 16


Médaille Copley
Tour Eiffel
Le Havre
3 juin
Baltimore
16 août
Académie frança
2 mars
26 janvier
29 novembre
28 mars
Iran
Londres
2006
1866
1939
Époque hellénistique
1809
IVe siècle av. J.-C.
1940
1998
1947

How to Embed Both Simultaneously?

- Build an “uber-graph” combining both graphs:

$$A' = \begin{bmatrix} A_E & B \\ C & A_F \end{bmatrix}$$

- What should C and B contain?
  - If they are 0, the two graphs embed orthogonally and we can do nothing.
  - If they are I (we assume the 1-1 correspondence of the E and F graphs, ordered this way) we get nearly orthogonal embeddings, still useless.
  - What if

$$B = C' = \frac{1}{2}(A_E + A_F)?$$

- The latter works. Will still need to discuss how to add a new document for which we do not know the match in the other language.

Random Projections

- It is well known that for many problems, one can obtain very good performance by simply choosing a random projection of high dimensional data.
- Is there a version of this for graphs? How can we choose a random projection of a graph into $\mathbb{R}^d$?
  - Let $Z \sim F$, and for each vertex i, draw z_i from F.
  - Project $v_i \to z_i \sum_{ij \in E} z_j$.
- The multiplier by z_i is so that this extends to hypergraphs, and so that cliques don't go to identical values.
- Repeat this d times to get an embedding into $\mathbb{R}^d$.
- Note: this is fast, no spectral calculations, can be done on huge graphs.

Outline

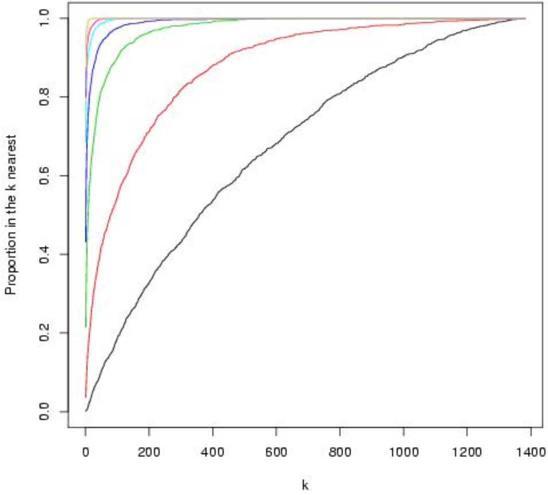
Evaluation

- Compare the distribution of distances between pairs to those between non-paired documents.
- Compute the proportion of times the paired document is within the top k nearest documents.

Algebraic Geometry Subset

- The data:
  - All articles in the English Wikipedia within two steps of the article “Algebraic Geometry”.
  - Corresponding French articles.
- The English and French graphs each contain 1382 vertices.

Algebraic Geometry k Nearest

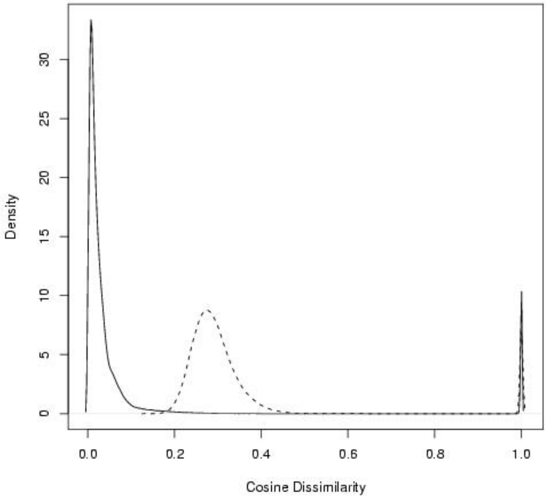


$d = 2, 5, 10, 25, 50, 100, 200.$

Full English and French Data

- All articles in the English Wikipedia for which a corresponding article in French exists and vice versa.
- The English and French graphs each contain $425,671$ vertices.

Full English and French Data Distributions



Outline

Glorified WCH

- The text can be incorporated via a bag-of-words approach by counting the number of times each word in the lexicon occurs in each document.
- These can be scaled via TFIDF or Mutual Information or other methods.
- These can be used directly to project the data – same issue of picking B, C, only now the A_E, A_F are the GWCH.

$$A' = \begin{bmatrix} A_E & B \\ C & A_F \end{bmatrix}$$

- Fill in B and C by averaging:
  - Find "close" documents (either in TFIDF space, or via the graph).
  - Average the GWCH from the associated pairs in the target language.
- Alternatively, we can use similarities in the GWCH to weight the B, C in the graph projection.
- Finally, we can project separately and combine the two projections (future work).

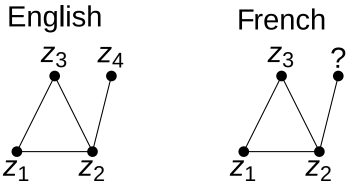
Outline

Embedding New Vertices

- How do we embed new vertices for which we don't known the associations?
  - Use the known associations:
    - if e denotes an English node, and f a French
    - we can look at e_i such that ee_i is an edge in the English graph
    - map these to corresponding French f_i
    - use these to "average" corresponding connections to fill in B and C.
- Use the GWCH to associate distances to the above, and do the same kind of "averaging" to fill in B and C entries.

Embedding New Vertices: Random Embedding

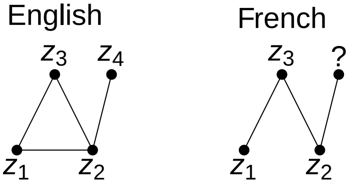
- How do we embed new vertices for which we don't known the associations?
- We want v_2 to go pretty much to the same place in English and French.



$$
\begin{aligned}
E &: \quad v_2 \mapsto z_2(z_1 + z_3 + z_4) \\
F &: \quad v_2 \mapsto z_2(z_1 + z_3 + ?) \\
\therefore & \quad v_4 \mapsto z_4
\end{aligned}
$$

Embedding New Vertices: Random Embedding

- How do we embed new vertices for which we don't known the associations?
- Solve an optimization problem mapping "pairs" as close as possible.



$$\begin{aligned}
E &: \quad v_2 \mapsto z_2(z_1 + z_3 + z_4) \\
F &: \quad v_2 \mapsto z_2(z_3 + ?) \\
\therefore & \quad v_4 \mapsto z_1 + z_4
\end{aligned}$$

Conclusions

- Multi-lingual Wikipediæ provide a nice testbed for various types of disparate data fusion experiments.
- Graph embeddings can be used to embed multiple graphs on the same actors into the same space.
- Spectral and random projections both provide useful methodologies worth pursuing.
- Random projections can be used on essentially arbitrarily large graphs.

Future Work

- Similar multi-lingual embedding using GWCH (language) ongoing.
- The optimization problem for random embedding of new vertices becomes much more complicated in real graphs. What is the "right" thing to do?
- Combine graph and GWCH.
- Can such embedding (where both documents and words are embedded) provide a start toward constructing dictionaries?
- Other approaches utilize dissimilarity matrices:
  - Convert the data into dissimilarity matrices.
  - Combine the dissimilarities.
  - Project.
- What are the properties of these different approaches, and when should one be used over another?