Envisioning Text

Learning Semantic Maps in Hyperbolic Space

Jörg Ontrup

Neuroinformatics Group
Faculty of Technology
Bielefeld University

Intelligent Information Systems 2005, Gdańsk
Outline of the Talk

1. Introduction
Outline of the Talk

1 Introduction

2 Methods
   - Self-organizing maps
   - Hyperbolic space
   - Learning of large maps with logarithmic complexity
   - The “bag of words” vs. the “pyramid of words”
Outline of the Talk

1. Introduction

2. Methods
   - Self-organizing maps
   - Hyperbolic space
   - Learning of large maps with logarithmic complexity
   - The “bag of words” vs. the “pyramid of words”

3. Applications
   - Organisation of large text archives
   - Handling of temporal data
   - An online demonstration
Outline of the Talk

1. Introduction
2. Methods
   - Self-organizing maps
   - Hyperbolic space
   - Learning of large maps with logarithmic complexity
   - The “bag of words” vs. the “pyramid of words”
3. Applications
   - Organisation of large text archives
   - Handling of temporal data
   - An online demonstration
4. Conclusion
40 years of Moore’s “Law”

- Exponential growth of transistors per integrated circuit
  ⇒ corresponding growth in computing power
  ⇒ equally important: exponential growth in storage capacities
Information Overload

40 years of Moore’s “Law”
- Exponential growth of transistors per integrated circuit
  ⇒ corresponding growth in computing power
  ⇒ equally important: exponential growth in storage capacities

Consequences & examples
- We observe an “explosion” of online available information
- Consider MEDLINE:
  - Citations from approx. 4800 journals
  - over 571,000 references added during 2004
  ⇒ 65 articles are added each hour
**Historical Notes - the Memex**

"As We May Think", Vannevar Bush, Juli 1945

- An essay in *Atlantic Monthly* draws attention
- *TIME Magazine* publishes several diagrams
Historical Notes - the Memex

“As We May Think”, Vannevar Bush, Juli 1945

- An essay in *Atlantic Monthly* draws attention
- *TIME Magazine* publishes several diagrams

“A memex is a device in which an individual stores all his books, records, and communications [...] It is an enlarged intimate supplement to his memory.”
Today’s Approaches

Classical information retrieval

- Massive full-text indexing of web sites
  ⇒ good when we know what to look for
Today’s Approaches

**Classical information retrieval**
- Massive full-text indexing of web sites
  ⇒ good when we know what to look for

**Machine learning methods**
- Clustering of query results
  ⇒ helps to identify topic clusters
## Today's Approaches

### Classical information retrieval
- Massive full-text indexing of web sites
  ⇒ good when we know what to look for

### Machine learning methods
- Clustering of query results
  ⇒ helps to identify topic clusters

### Information Visualization
- Often used: map metaphor
  ⇒ allows navigation in information space
Future Directions

The Semantic Web

- From pure syntax to semantic information
- Semantic annotations lessen the burden to “hit” the right keyword
Future Directions

The Semantic Web

- From pure syntax to semantic information
- Semantic annotations lessen the burden to “hit” the right keyword

Still quite a way to go:

- Manual tagging of content is very expensive
- Hard to construct ontologies dealing with general knowledge
- Even harder to do that automatically
How is our Brain Organizing Information?

- Consider input from our senses, e.g. the sense of touch:
How is our Brain Organizing Information?

- Consider input from our senses, e.g. the sense of touch:
How is our Brain Organizing Information?

- Consider input from our senses, e.g. the sense of touch:

The Self-Organizing Map as a Model

- Introduced by Kohonen in the 80s
- Biophysically motivated model
How is our Brain Organizing Information?

Consider input from our senses, e.g. the sense of touch:

The Self-Organizing Map as a Model

- Introduced by Kohonen in the 80s
- Biophysically motivated model
  - unsupervised learning paradigm
  - generates *topologically ordered* feature maps
How is our Brain Organizing Information?

Consider input from our senses, e.g. the sense of touch:

The Self-Organizing Map as a Model

- Introduced by Kohonen in the 80s
- Biophysically motivated model
  - unsupervised learning paradigm
  - generates *topologically ordered* feature maps
- Translates *data similarities* into *spatial relations*
- Widely used as an exploratory data analysis tool
Self-Organizing Maps

- Cortex is modelled by neurons placed on lattice structure
- Most prominently used are grids in 2D Euclidean space
Self-Organizing Maps

- Cortex is modelled by neurons placed on lattice structure
- Most prominently used are grids in 2D Euclidean space
- SOM algorithm:

High-dimensional input space

Lattice of neurons
Self-Organizing Maps

- Cortex is modelled by neurons placed on lattice structure
- Most prominently used are grids in 2D Euclidean space
- SOM algorithm:
Self-Organizing Maps

- Cortex is modelled by neurons placed on lattice structure
- Most prominently used are grids in 2D Euclidean space
- SOM algorithm:
Self-Organizing Maps

- Cortex is modelled by neurons placed on lattice structure
- Most prominently used are grids in 2D Euclidean space
- SOM algorithm:
Self-Organizing Maps

<table>
<thead>
<tr>
<th>animal</th>
<th>small</th>
<th>medium</th>
<th>big</th>
<th>2 legs</th>
<th>hooves</th>
<th>hunter</th>
<th>can fly</th>
<th>swimmer</th>
</tr>
</thead>
<tbody>
<tr>
<td>cow</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>zebra</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>horse</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>lion</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>tiger</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>cat</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>wolf</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>dog</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>fox</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>eagle</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>hawk</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>owl</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>goose</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>duck</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>hen</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>dove</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Self-Organizing Maps

- horse/zebra
- lion
- dog
- wolf
- fox
- cat
- hen
- eagle
- dove
- owl/hawk
- goose
- duck
SOM-based Document Maps

- The WebSOM project with 7 million patent abstracts
SOM-based Document Maps

- The WebSOM project with 7 million patent abstracts
- André Skupin applied cartographic design concepts:
Pros and Cons of the standard SOM

**Strengths**

- Completely unsupervised method
- Provides map metaphor well suited for visualization purposes
- Achieves good classification rates, can capture non-linearities
- Has proven its applicability in many domains
Pros and Cons of the standard SOM

**Strengths**
- Completely unsupervised method
- Provides map metaphor well suited for visualization purposes
- Achieves good classification rates, can capture non-linearities
- Has proven its applicability in many domains

**Limitations**
- Resolution depends on map area $A \propto N^2$
  - Training of large maps is computationally expensive
- Large maps do not fit on limited screens
  - Demand for focus & context techniques
Going to Hyperbolic Space

- World of Euclidean geometry is following Euclid’s *five axioms*
- Euclid’s 5th or “parallel postulate” was long eyed suspiciously:
Going to Hyperbolic Space

- World of Euclidean geometry is following Euclid’s *five axioms*
- Euclid’s 5th or “parallel postulate” was long eyed suspiciously:

\[
\begin{align*}
\text{Through a given point, not on a} \\
\text{given line}
\end{align*}
\]
Going to Hyperbolic Space

- World of Euclidean geometry is following Euclid’s five axioms
- Euclid’s 5th or “parallel postulate” was long eyed suspiciously:

Through a given point, not on a given line only one parallel can be drawn to the given line.
Going to Hyperbolic Space

- World of Euclidean geometry is following Euclid’s *five axioms*
- Euclid’s 5th or “parallel postulate” was long eyed suspiciously:

\[ P \]

Through a given point, not on a given line only one parallel can be drawn to the given line.

- Around 1820 Gauss, Lobachevsky & Bolyai independently negated the 5th postulate
Going to Hyperbolic Space

- World of Euclidean geometry is following Euclid’s *five axioms*
- Euclid’s 5th or “parallel postulate” was long eyed suspiciously:

  ![Diagram](image.png)

  *Through a given point, not on a given line only one parallel can be drawn to the given line.*

- Around 1820 Gauss, Lobachevsky & Bolyai independently negated the 5th postulate
  ⇒ They found a completely consistent non-Euclidean geometry
  ⇒ characterized by being negatively “curved”
Going to Hyperbolic Space

How does the hyperbolic plane $H^2$ look like?

- Locally isometric embedding in $R^3$
  - yields “wrinkled” structure
  - resembles a saddle at every point of surface
Going to Hyperbolic Space

How does the hyperbolic plane $H^2$ look like?

- Locally isometric embedding in $\mathbb{R}^3$
  - yields “wrinkled” structure
  - resembles a saddle at every point of surface

Characteristics of Hyperbolic Space

- For area $A$ and circumference $C$ of circle with radius $r$ holds:
  \[ A(r) = 4\pi \sinh^2\left(\frac{r}{2}\right), \quad C(r) = 2\pi \sinh(r) \]
  - $A$ and $C$ grow asymptotically exponentially with the radius
Projection of $\mathbb{H}^2$

- How can we project infinite plane $\mathbb{H}^2$ onto Euclidean screen?
Projection of $H^2$

- How can we project infinite plane $H^2$ onto Euclidean screen?

The Poincaré Disk

- Under embedding of $H^2$ in Minkowski-Space

\[
\begin{align*}
  x &= \sinh(\rho) \cos(\phi) \\
  y &= \sinh(\rho) \sin(\phi) \\
  u &= \cosh(\rho)
\end{align*}
\]

$H^2$ appears as surface M which can be projected to the unit disk
The Poincaré Disk

Properties of the Poincaré Modell

- Locally shape preserving
- Non-isometric with strong “fish-eye” effect:
  - Fovea is mapped faithfully
  - Distant regions are “squeezed”
- Möbius transformation allows elegant translation of fovea
The Poincaré Disk

Properties of the Poincaré Modell

- Locally shape preserving
- Non-isometric with strong “fish-eye” effect:
  - Fovea is mapped faithfully
  - Distant regions are “squeezed”
- Möbius transformation allows elegant translation of fovea
The Poincaré Disk

Properties of the Poincaré Modell

- Locally shape preserving
- Non-isometric with strong “fish-eye” effect:
  - Fovea is mapped faithfully
  - Distant regions are “squeezed”
- Möbius transformation allows elegant translation of fovea
The Poincaré Disk

Properties of the Poincaré Modell

- Locally shape preserving
- Non-isometric with strong “fish-eye” effect:
  - Fovea is mapped faithfully
  - Distant regions are “squeezed”
- Möbius transformation allows elegant translation of fovea
The Hyperbolic Self-Organizing Map

- Has been introduced by Helge Ritter in 1999
- Nodes are placed on vertices of regular tessellation of $H^2$ with equilateral triangles
The Hyperbolic Self-Organizing Map

- Has been introduced by Helge Ritter in 1999
- Nodes are placed on vertices of regular tesselation of $H^2$ with equilateral triangles

Pros

- Shows favourable classification results
- Allows for intuitive browsing
The Hyperbolic Self-Organizing Map

- Has been introduced by Helge Ritter in 1999
- Nodes are placed on vertices of regular tessellation of $H^2$ with equilateral triangles

**Pros**
- Shows favourable classification results
- Allows for intuitive browsing

**Cons**
- Training of very large maps still expensive
- Most things “happen” at the perimeter
Architecture of the $H^2$SOM

1. Initialization:
   - start with $n_b$ equilateral triangles
   - place nodes at outer vertices
Architecture of the $H^2$SOM

1. **Initialization:**
   - start with $n_b$ equilateral triangles
   - place nodes at outer vertices
   - train nodes with traditional SOM approach:

\[
\Delta w_a = \epsilon(t) h(a, a^*) (x - w_a)
\]

\[
h(a, a^*) = \exp \left( -\frac{d_{a,a^*}^2}{\sigma(t)^2} \right)
\]

\[
d_{a,a^*} = 2 \arctanh \left( \frac{|z_a - z_{a^*}|}{|1 - z_a \bar{z}_{a^*}|} \right)
\]

- nodes now form 1st level of hierarchy
Architecture of the $H^2$SOM

2 Growth Step:

- compute quantization error for nodes & mark those with $QE > \Theta_{QE}$
Architecture of the $H^2$SOM

2 Growth Step:
- compute quantization error for nodes & mark those with $QE > \Theta_{QE}$
- expand nodes with $QE > \Theta_{QE}$
Architecture of the H<sup>2</sup>SOM

2 Growth Step:
- compute quantization error for nodes & mark those with $QE > \Theta_{QE}$
- expand nodes with $QE > \Theta_{QE}$
- fixate neurons from inner hierarchies & “move” adaptation to new structural level
Architecture of the H^2SOM

2 Growth Step:
- compute quantization error for nodes & mark those with \( QE > \Theta_{QE} \)
- expand nodes with \( QE > \Theta_{QE} \)
- fixate neurons from inner hierarchies & “move” adaptation to new structural level
- reiterate growing step
Architecture of the $H^2$SOM

**Growth Step:**
- compute quantization error for nodes & mark those with $QE > \Theta_{QE}$
- expand nodes with $QE > \Theta_{QE}$
- fixate neurons from inner hierarchies & “move” adaptation to new structural level
- reiterate growing step
Architecture of the $H^2$SOM

2 Growth Step:
- compute quantization error for nodes & mark those with $QE > \Theta_{QE}$
- expand nodes with $QE > \Theta_{QE}$
- fixate neurons from inner hierarchies & “move” adaptation to new structural level
- reiterate growing step
Architecture of the $H^2$SOM

2 Growth Step:
- compute quantization error for nodes & mark those with $QE > \Theta_{QE}$
- expand nodes with $QE > \Theta_{QE}$
- fixate neurons from inner hierarchies & “move” adaptation to new structural level
- reiterate growing step
Fast Tree Search

“Uniformly hierarchical” structure of hyperbolic grid allows significant speed-up for finding the winner neuron:
Fast Tree Search

“Uniformly hierarchical” structure of hyperbolic grid allows significant speed-up for finding the winner neuron:

⇒ for $k = 1$ we need $O(\log_{nb} N)$ comparisons, instead of $O(N)$

⇒ Training of very large maps with logarithmic complexity
Fast Tree Search

“Uniformly hierarchical” structure of hyperbolic grid allows significant speed-up for finding the winner neuron:

Benchmark with MNIST database of handwritten digits
- training time: 13 min vs. 18h 34min
- classification: 94.6% vs. 92.7%
Text Representation

- Standard approach for text representation: “bag of words”
Alternative: use WordNet and its hypernym hierarchy

⇒ We might call it a “pyramid of words”
Text Representation

⇒ WordNet allows for a hierarchical representation of text data:

```
class Vegetable:
    def __init__(self, name):
        self.name = name

class Fruit:
    def __init__(self, name):
        self.name = name

class Alcohol:
    def __init__(self, name):
        self.name = name

tomato = Vegetable("tomato")
aubergine = Vegetable("aubergine")
strawberry = Fruit("strawberry")
raspberry = Fruit("raspberry")
merlot = Alcohol("merlot")
vodka = Alcohol("vodka")
```

```
wordnet = WordNet

wordnet.add_concept(Vegetable("tomato"))
wordnet.add_concept(Vegetable("aubergine"))
wordnet.add_concept(Fruit("strawberry"))
wordnet.add_concept(Fruit("raspberry"))
wordnet.add_concept(Alcohol("merlot"))
wordnet.add_concept(Alcohol("vodka"))
```
Hierarchical Representation & Training

- Hierarchical data representation ideally suited for hierarchical maps

⇒ train the map using the “pyramid of words”:

- food
- fruit
- vegetable
- alcohol
- tomato
- aubergine
- strawberry
- raspberry
- merlot
- chianti
- vodka
Organisation of Large Text Archives

1. Initial map is trained with complete document set
Organisation of Large Text Archives

2. First structural level partitions dataset into subclusters
Organisation of Large Text Archives

Nodes are expanded and trained by fast tree search approach
Organisation of Large Text Archives

Growing network splits documents in topological ordered clusters of decreasing size
Organisation of Large Text Archives

Growing network splits documents in topological ordered clusters of decreasing size.
Event Detection in Document Streams

Many documents contain temporal information

- News feeds & tickers such as Reuters
- Chatroom messages & web forums
- Mailing lists or incoming e-mails
Event Detection in Document Streams

Many documents contain temporal information
- News feeds & tickers such as Reuters
- Chatroom messages & web forums
- Mailing lists or incoming e-mails

Attach time dependent activation potential to HSOM’s neurons

\[ a_i(t) = \beta a_i(t - 1) + S_i(t) \]

\[ S_i(t) = \begin{cases} 1 & \text{if } i \text{ is best-match node at arrival time } t \\ 0 & \text{otherwise} \end{cases} \]

\( \beta \): decay factor controlling amount of leakage
Demo
Demo

Animate Focus

2002-10-16 00:08:00
Demo

Animate Focus

2002-10-18 00:01:00
Summary

- Hyperbolic space offers more freedom to build hierarchical maps
- The peculiar geometry allows for focus & context enhanced browsing
- Large maps of documents can be build with logarithmic complexity
Thank you very much!