Nycomed Chair for
Bioinformatics and Information Mining

Universität
Konstanz

# Pure Spreading Activation is pointless

11. 09. 2009

Michael R. Berthold, Ulrik Brandes, Tobias Kötter, Martin Mader, Uwe Nagel, Kilian Thiel

# Agenda

- Spreading Activation
  - Introduction
  - Framework
- Query-Independence
- Avoiding Query-Independence
- Application
- Results
- Conclusion and Outlook

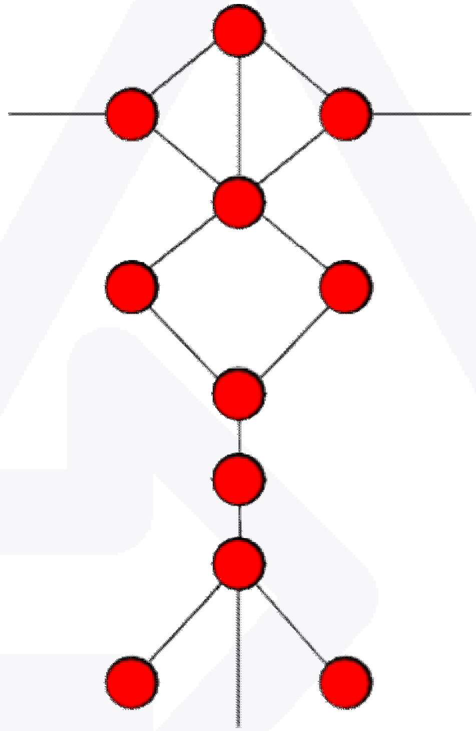
What is it used for ?

- To query graphs
- Find interesting / relevant
  - units of information (nodes)
  - relations of such units (edges)
- Rank units according to their degree of activation

# Spreading Activation

How does Spreading Activation typically work ?

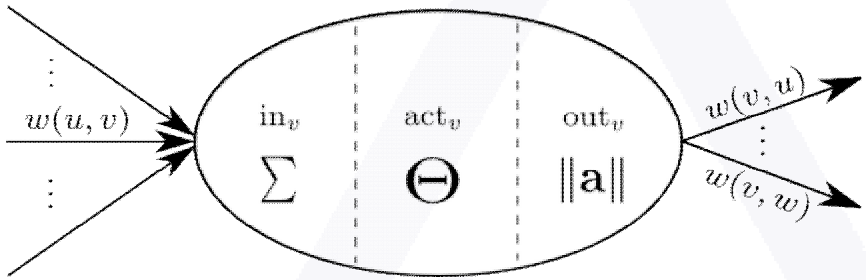
- Iterative process
- Initially activate nodes
- In each iteration:
  - Distribute activation along incident (weighted) edges
  - Combine incoming activation at target nodes
- Stop iterative process when:
  - System converges
  - Constraints
    - Maximum number of activated nodes / edges
    - Maximum number of iterations
    - …
- Return activated nodes and edges as result
- Rank nodes by their activation



# Framework

Related to neural networks (Rumelhart et al.) a tripartition of spreading activation is proposed:

- Input
  - Function $\mathbf{in}_v : \mathbb{R}^m \to \mathbb{R}$
  - State $\mathbf{i}_v^{(t)} = \mathrm{in}_v(\mathbf{o}^{(t-1)})$
  - Combines the incoming activation




- Activation
  - Function $\mathrm{act}_v : \mathbb{R} \to \mathbb{R}$
  - State $\mathbf{a}_v^{(t)} = \mathrm{act}_v(\mathbf{i}_v^{(t)})$
  - Determines the activation of a node (according to the input)
  - Adds non linearity

- Output
  - Function $\mathrm{out}_v : \mathbb{R} \to \mathbb{R}$
  - State $\mathbf{o}_v^{(t)} = \mathrm{out}_v(\mathbf{a}_v^{(t)})$
  - Computes the outgoing activation (according to the activation)
  - Normalization, etc.

# Pure Linear Spreading Activation

Linear Spreading Activation framework:

- Input: $$\mathbf{i}_v^{(t)} = \sum_{u \in N(v)} \mathbf{o}_u^{(t-1)} w(u,v)$$
- Activation: $$\mathbf{a}_v^{(t)} = \mathbf{i}_v^{(t)}$$
- Output: $$\mathbf{o}_v^{(t)} = \mathbf{a}_v^{(t)}$$
  - Simplified: $$\mathbf{a}_v^{(t)} = \sum_{u \in N(v)} \mathbf{a}_u^{(t-1)} w(u,v)$$
- With the weighted adjacency matrix $W$ of a given Graph this yields (non normalized) to:

$$\mathbf{a}^{(t)} = W\mathbf{a}^{(t-1)} \quad \text{or} \quad \mathbf{a}^{(t)} = W^t \mathbf{a}^{(0)}$$

and normalized: $$\mathbf{a}^{(t)} = \frac{W^t \mathbf{a}^{(0)}}{\|W^t \mathbf{a}^{(0)}\|}$$

# Convergence of pure Spreading Activation

- The direction of $\mathbf{a}^{(t)}$ converges against the (direction of the) eigenvector $\mathbf{x}_1$ of the eigenwert $e_1$ of $W$ with the maximum absolute value (under some conditions) for increasing t's (w.r.t. $\left|\frac{e_2}{e_1}\right|$ ) [Perron-Froebenius].
  - Order of nodes sorted by their activation depends on the direction of $\mathbf{a}^{(t)}$
  - Query independent result (not suitable)
  - Represents a global answer to all queries
- A mechanism is necessary which prevents convergence to one fixed point

# Spreading Activation result handling

Spreading Activation policies and result handling methods:

– Accumulation $$\mathbf{a} = \sum_{t=0}^{\infty} \lambda(t)\mathbf{a}^{(t)}$$

– Activation renewing $$\mathbf{a}^{(t)} = \mathbf{a}^{(0)} + W\mathbf{a}^{(t-1)}$$

– Inertia $$\mathbf{a}^{(t)} = \mathbf{a}^{(t-1)} + W\mathbf{a}^{(t-1)}$$

# Accumulation

Accumulation of iteration results with a decay $\lambda(t)$ for each iteration

$$\mathbf{a} = \sum_{t=0}^{\infty} \lambda(t)\mathbf{a}^{(t)} = \sum_{t=0}^{\infty} \lambda(t) W^t \mathbf{a}^{(0)}$$

- For $\lambda(t) = \alpha^t$ and $0 < \alpha < \frac{1}{e_1}$, with $e_1$ as eigenwert of $W$ with maximum absolute value of, this converges to [Katz]:

  $$\mathbf{a} = (I - \alpha W)^{-1}\mathbf{a}^{(0)}$$

- The accumulation enables convergence to a fixed point dependent on the query
- The decay factor controls the impact of each additional iteration
  - First iterations represent local results, according to the query
  - Fixed point represents global result

# Activation Renewing

Renewing of the initial activation in each iteration

$$\mathbf{a}^{(t)} = \mathbf{a}^{(0)} + W\mathbf{a}^{(t-1)}$$

which leads to

$$\mathbf{a}^{(t)} = \left( \sum_{i=0}^{t} W^i \right) \mathbf{a}^{(0)}$$

- Renewing of activation converges only if $e_1 < 1$

- To increase the impact of the initially activated vertices their activation is renewed in each iteration.
- Convergence is not assured for all networks.

# Inertia

Conservation of the last activation in each iteration

$$\mathbf{a}^{(t)} = \mathbf{a}^{(t-1)} + W\mathbf{a}^{(t-1)} = (I + W)^t \mathbf{a}^{(0)}$$

– Cumulating all iterations with a decay $\lambda(t) = \alpha^t$ leads to

$$\mathbf{a} = ((1-\alpha)I - \alpha W)^{-1} \mathbf{a}^{(0)}$$

– Conservation of activation converges only for $0 < \alpha < \frac{1}{e_1}$ with $e_1$ as the eigenwert of $(I + W)$ with the maximum absolute value.
– Similar to spreading activation on $W$ but with self loops.

# Example Application

To confirm the convergence behavior of Spreading Activation an Information Retrieval application has been applied.

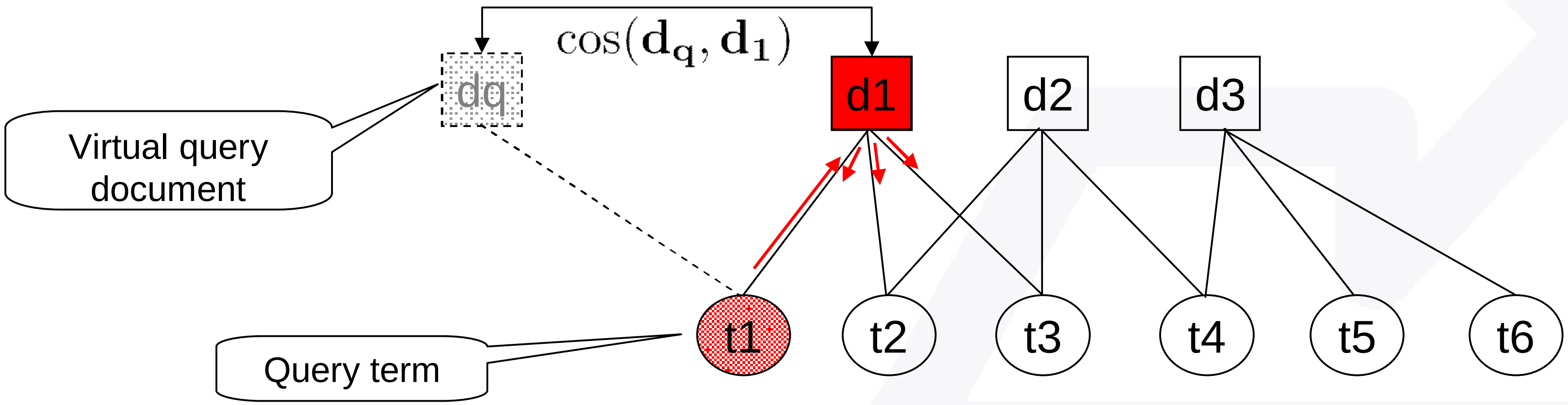
- TIME and MED part of the Smart test collection as benchmark data sets

| Test collection | Number of documents | Number of queries |
| --- | --- | --- |
| TIME | 425 | 83 |
| MED | 1033 | 30 |

- The network consists of a bipartite document-term graph, with weighted edges (TFIDF).
- with $W \in \mathbb{R}^{m \times n}$ as the adjacency matrix of the document-term graph,
- the document vectors $\mathbf{d}_i \in \mathbb{R}^n$, $1 \leq i \leq m$
- the term vectors $\mathbf{t}_j \in \mathbb{R}^m$, $1 \leq j \leq n$

# Example Application

- Cosine is a popular similarity measure in IR
- Alternating Cosine
  - Activation of a document represents the cosine similarity to the previous virtual document.
  - Activation of a term represents the cosine similarity to the previous virtual term.
  - The resulting documents are sorted by the activation of the corresponding vertex

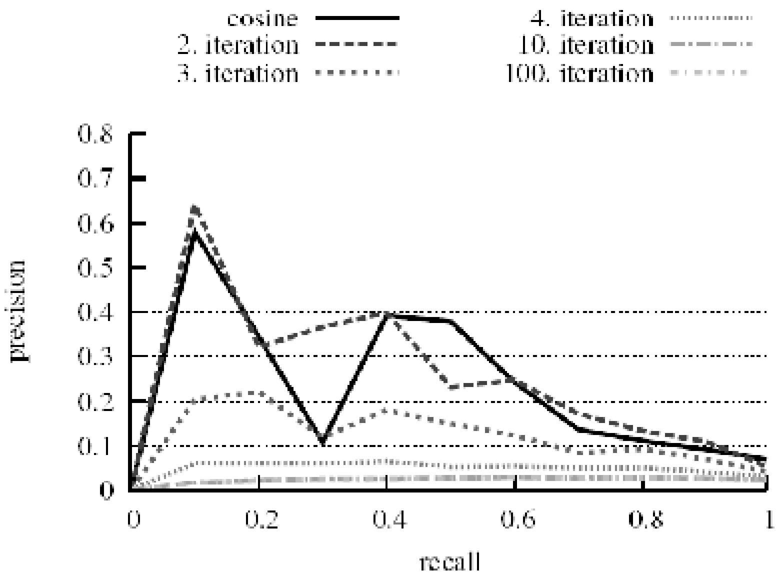


# Example Application

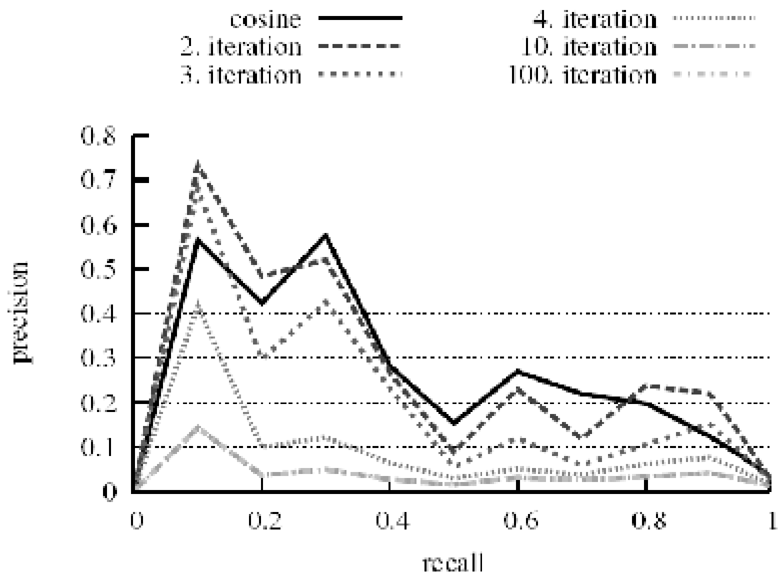
Results of pure iterations

- Precision measured at 11 recall values
- Average precision of all queries

MED




TIME




Converges after ca. 10 iterations with very low precision values.

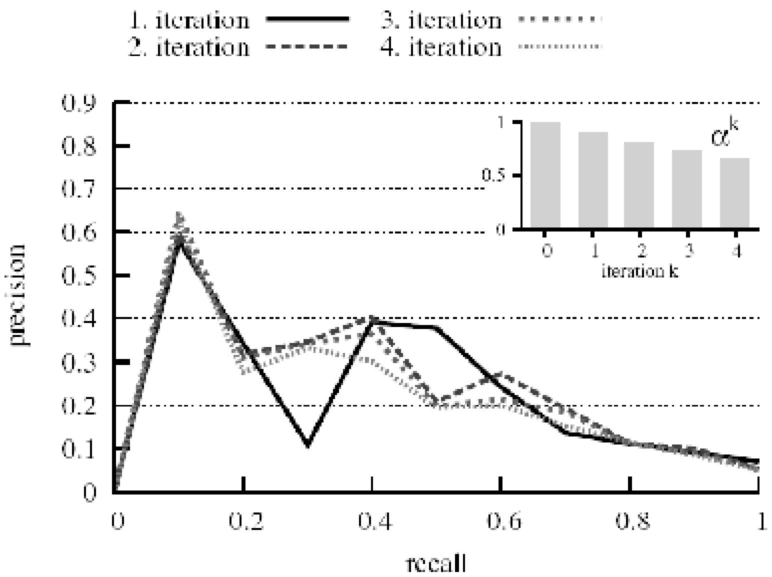
- Fixed point is an inadequate result

# Example Application

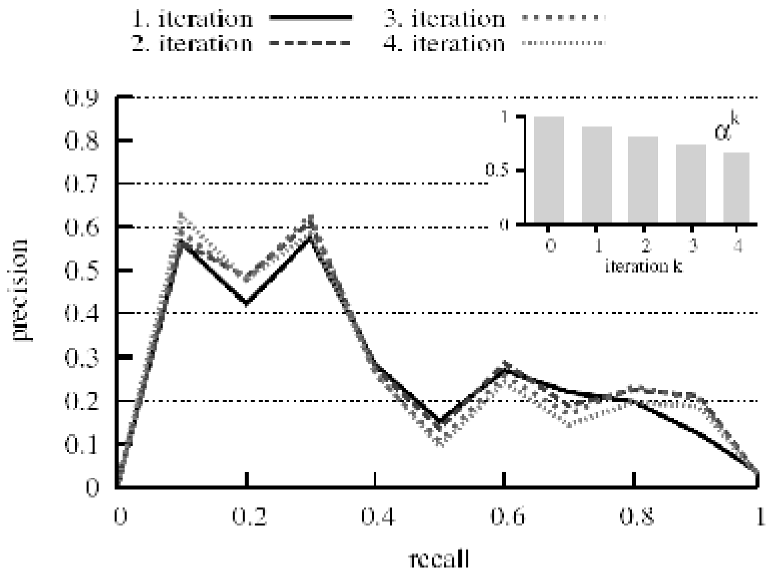
Results of accumulated iterations

- Decay $\alpha = 0.9$
- Precision measured at 11 recall values
- Average precision of all queries

MED




TIME




- Fixed points are more reasonable results

# Conclusion

- Pure linear Spreading Activation converges against a query independent fixed point.
  - Not adequate to answer queries.
  - Approaches to avoid query-independence:
    - Constraints (hard to analyze)
    - Accumulation of iteration results etc.

- Accumulation decay/function regulates local/global characteristic of the final result.

# Outlook

- Apply Spreading Activation on BisoNets in order to
  - find possible bisociations related to a topical environment
  - find relevant information to answer a query (local answer)
  - find interesting information to support creativity (more global answer)

- Partitioned graphs
  - For each partition can be defined
    - Input, activation, and output function
    - Decay value, function

- What kind of Spreading Activation Framework allows to find reasonable results ?
  - Non linear activation functions
    - Convergence behavior
    - Number of fixed points / attractors
  - BFS vs. DFS

# Thank You