Chapter 2
Modeling

Set Theoretic Models
Fuzzy Set Model
Extended Boolean Model
The Generalized Vector Model
Latent Semantic Indexing
Neural Network for IR
Set Theoretic Models
Set Theoretic Models

- The Boolean model imposes a binary criterion for deciding relevance
- The question of how to extend the Boolean model to accommodate partial matching, i.e., a ranking for the documents retrieved has attracted considerable attention in the past
- We discuss now two set theoretic models for this:
  - Fuzzy Set Model
  - Extended Boolean Model
Fuzzy Set Model
Fuzzy Set Model

Queries and docs represented by sets of index terms: matching is *approximate* from the start.

This *vagueness* can be modeled using a fuzzy framework, as follows:

- with each term is associated a *fuzzy* set
- each doc has a degree of membership in this fuzzy set

This interpretation provides the foundation for many IR models based on fuzzy theory.

In here, we discuss the model proposed by Ogawa, Morita, and Kobayashi (1991).
Fuzzy Set Theory

- Framework for representing classes whose boundaries are not well defined
- Key idea is to introduce the notion of a degree of membership associated with the elements of a set
- This degree of membership varies from 0 to 1 and allows modeling the notion of marginal membership
- Thus, membership is now a gradual notion, contrary to the crispy notion enforced by classic Boolean logic
Fuzzy Set Theory

Definition

A fuzzy subset $A$ of a universe of discourse $U$ is characterized by a membership function

$$\mu_A : U \rightarrow [0, 1]$$

which associates with each element $u$ of $U$ a number $\mu_A(u)$ in the interval $[0,1]$.

Definition

Let $U$ be the universe of discourse, $A$ and $B$ be two fuzzy subsets of $U$, and $\overline{A}$ be the complement of $A$ relative to $U$. Also, let $u$ be an element of $U$. Then,

$$\mu_{\overline{A}}(u) = 1 - \mu_A(u)$$

$$\mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u))$$

$$\mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u))$$
Fuzzy Information Retrieval

- Fuzzy sets are modeled based on a thesaurus
- This thesaurus is built as follows:
  - Let $\vec{c}$ be a term-term correlation matrix
  - Let $c_{i,l}$ be a normalized correlation factor between two terms $k_i$ and $k_l$:
    \[
    c_{i,l} = \frac{n_{i,l}}{n_i + n_l - n_{i,l}}
    \]
    - $n_i$: number of docs which contain $k_i$
    - $n_l$: number of docs which contain $k_l$
    - $n_{i,l}$: number of docs which contain both $k_i$ and $k_l$
- We now have the notion of *proximity* among index terms.
The correlation factor $c_{i,l}$ can be used to define fuzzy set membership for a document $d_j$ as follows:

$$
\mu_{i,j} = 1 - \prod_{k_l \in d_j} (1 - c_{i,l})
$$

$\mu_{i,j}$: membership of doc $d_j$ in fuzzy subset associated with $k_i$

The above expression computes an algebraic sum over all terms in $d_j$

A document $d_j$ belongs to the fuzzy set associated with $k_i$, if its own terms are associated with $k_i$
If $d_j$ contains a term $k_l$ which is closely related to $k_i$, we have

- $c_{i,l} \sim 1$
- $\mu_{i,j} \sim 1$
- and $k_i$ is a good fuzzy index for $d_j$

$$
\mu_{i,j} = 1 - \prod_{k_l \in d_j} (1 - c_{i,l})
$$

$\mu_{i,j}$ : membership of doc $d_j$ in fuzzy subset associated with $k_i$
Disjunct normal form is given by

\[
\vec{q}_{dnf} = (1, 1, 1) + (1, 1, 0) + (1, 0, 0) = cc_1 + cc_2 + cc_3
\]
Fuzzy IR: An Example

\[ \mu_{q,j} = \mu_{cc_1+cc_2+cc_3,j} \]

\[ = 1 - \prod_{i=1}^{3} (1 - \mu_{cc_i,j}) \]

\[ = 1 - (1 - \mu_{a,j} \mu_{b,j} \mu_{c,j}) \times \\
(1 - \mu_{a,j} \mu_{b,j} (1 - \mu_{c,j})) \times (1 - \mu_{a,j} (1 - \mu_{b,j})(1 - \mu_{c,j})) \]
Fuzzy Information Retrieval

- Fuzzy IR models have been discussed mainly in the literature associated with fuzzy theory
- Experiments with standard test collections are not available
- Difficult to compare at this time
Extended Boolean Model
Extended Boolean Model

- Boolean model is simple and elegant
- But, no provision for a ranking
- As with the fuzzy model, a ranking can be obtained by relaxing the condition on set membership
- Extend the Boolean model with the notions of partial matching and term weighting
- Combine characteristics of the Vector model with properties of Boolean algebra
The Idea

The extended Boolean model (introduced by Salton, Fox, and Wu, 1983) is based on a critique of a basic assumption in Boolean algebra

Let,

\[ q = k_x \land k_y \]

\[ w_{x,j} = f_{x,j} \times \frac{idf_x}{\max_i idf_i} \]

\[ w_{x,j} \text{ is a weight associated with the pair } [k_x, d_j] \]

To simplify notation, let

\[ w_{x,j} = x \text{ and } w_{y,j} = y \]
The Idea

\[ \text{sim}(q_{or}, d) = \sqrt{x^2 + y^2} \]

\[ \text{sim}(q_{and}, d) = 1 - \sqrt{(1 - x)^2 + (1 - y)^2} \]
Generalizing the Idea

We can extend the previous model to consider Euclidean distances in a t-dimensional space.

This can be done using $p$-norms which extend the notion of distance to include $p$-distances, where $1 \leq p \leq \infty$ is a new parameter.

A generalized conjunctive query is given by

$$q_{\text{and}} = k_1 \land^p k_2 \land^p \ldots \land^p k_m$$

A generalized disjunctive query is given by

$$q_{\text{or}} = k_1 \lor^p k_2 \lor^p \ldots \lor^p k_m$$
Generalizing the Idea

The query-document similarities are now given by

\[
sim(q_{or}, d_j) = \left( \frac{x_1^p + x_2^p + \ldots + x_m^p}{m} \right)^{\frac{1}{p}}
\]

\[
sim(q_{and}, d_j) = 1 - \left( \frac{(1 - x_1)^p + (1 - x_2)^p + \ldots + (1 - x_m)^p}{m} \right)^{\frac{1}{p}}
\]

where each \( x_i \) stands for the weight \( w_{i,d} \) associated to the pair \([k_i, d_j]\).
Properties

\[ \text{sim}(q_{or}, d_j) = \left( \frac{x_1^p + x_2^p + \ldots + x_m^p}{m} \right) \frac{1}{p} \]

\[ \text{sim}(q_{and}, d_j) = 1 - \left( \frac{(1-x_1)^p + (1-x_2)^p + \ldots + (1-x_m)^p}{m} \right) \frac{1}{p} \]

If \( p = 1 \) then (vector-like)

\[ \text{sim}(q_{or}, d_j) = \text{sim}(q_{and}, d_j) = \frac{x_1 + \ldots + x_m}{m} \]

If \( p = \infty \) then (Fuzzy like)

\[ \text{sim}(q_{or}, d_j) = \max(x_i) \]

\[ \text{sim}(q_{and}, d_j) = \min(x_i) \]
Properties

- By varying $p$, we can make the model behave as a vector, as a fuzzy, or as an intermediary model.

- This is quite powerful and is a good argument in favor of the extended Boolean model.

$\begin{align*}
q &= (k_1 \land^p k_2) \lor^p k_3 \\
&= (k_1 \land^p k_2) \lor^p k_3
\end{align*}$

- $k_1$ and $k_2$ are to be used as in a vectorial retrieval while the presence of $k_3$ is required.

$\begin{align*}
sim(q, d) &= \left( \frac{1 - \left( \frac{(1-x_1)^p + (1-x_2)^p}{2} \right)^\frac{1}{p} + x_3^p}{2} \right)^\frac{1}{p}
\end{align*}$
Conclusions

- Model is quite powerful
- Properties are interesting and might be useful
- Computation is somewhat complex
- However, distributivity operation does not hold for ranking computation:
  \[ q_1 = (k_1 \lor k_2) \land k_3 \]
  \[ q_2 = (k_1 \land k_3) \lor (k_2 \land k_3) \]
  \[ \text{sim}(q_1, d_j) \neq \text{sim}(q_2, d_j) \]
Algebraic Models
Generalized Vector Model
Generalized Vector Model

- Classic models enforce independence of index terms
- For the Vector model:
  - Set of term vectors \{ \vec{k}_1, \vec{k}_2, \ldots, \vec{k}_t \} are linearly independent and form a basis for the subspace of interest
- Frequently, this is interpreted as:
  \[ \forall i, j \Rightarrow \vec{k}_i \cdot \vec{k}_j = 0 \]
- In 1985, Wong, Ziarko, and Wong proposed an interpretation in which the set of terms is linearly independent, but not pairwise orthogonal
In the generalized vector model, two index terms might be non-orthogonal and are represented in terms of smaller components (minterms).

As before let,

- \( w_{i,j} \) be the weight associated with \([k_i, d_j]\)
- \( \{k_1, k_2, \ldots, k_t\} \) be the set of all terms

If these weights are all binary, all patterns of occurrence of terms within documents can be represented by the minterms:

- \( m_1 = (0, 0, \ldots, 0) \), \( m_2 = (1, 0, \ldots, 0) \), \ldots,
- \( m_{2^t} = (1, 1, \ldots, 1) \)

In here, \( m_2 \) indicates documents in which solely the term \( k_1 \) occurs.
Key Idea

The basis for the generalized vector model is formed by a set of $2^t$ vectors defined over the set of minterms, as follows:

\[ \vec{m}_1 = (1, 0, \ldots, 0, 0) \]
\[ \vec{m}_2 = (0, 1, \ldots, 0, 0) \]
\[ \vdots \]
\[ \vec{m}_t = (0, 0, \ldots, 0, 1) \]

Notice that, \[ \forall i, j \Rightarrow \vec{m}_i \cdot \vec{m}_j = 0 \] i.e., pairwise orthogonal
Minterm vectors are pairwise orthogonal. But, this does not mean that the index terms are independent:

The minterm $m_4$ is given by:

$$m_4 = (1, 1, 0, \ldots, 0)$$

This minterm indicates the occurrence of the terms $k_1$ and $k_2$ within a same document. If such document exists in a collection, we say that the minterm $m_4$ is active and that a dependency between these two terms is induced.

The generalized vector model adopts as a basic foundation the notion that co-occurrence of terms within documents induces dependencies among them.
Forming the Term Vectors

The vector associated with the term $k_i$ is computed as:

$$
\vec{k}_i = \frac{\sum \forall r, g_i(m_r)=1 c_{i,r} \vec{m}_r}{\sqrt{\sum \forall r, g_i(m_r)=1 c_{i,r}^2}}
$$

$$
c_{i,r} = \sum_{d_j \mid g_l(\vec{d}_j)=g_l(m_r) \text{ for all } l} w_{i,j}
$$

The weight $c_{i,r}$ associated with the pair $[k_i, m_r]$ sums up the weights of the term $k_i$ in all the documents which have a term occurrence pattern given by $m_r$.

Notice that for a collection of size $N$, only $N$ minterms affect the ranking (and not $2^t$)
Dependency between Index Terms

A degree of correlation between the terms $k_i$ and $k_j$ can now be computed as:

$$\vec{k}_i \cdot \vec{k}_j = \sum_{\forall r \mid g_i(m_r) = 1 \land g_j(m_r) = 1} c_{i,r} \times c_{j,r}$$

This degree of correlation sums up (in a weighted form) the dependencies between $k_i$ and $k_j$ induced by the documents in the collection (represented by the $m_r$ minterms).
The Generalized Vector Model

An Example

<table>
<thead>
<tr>
<th></th>
<th>$K_1$</th>
<th>$K_2$</th>
<th>$K_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_1$</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$d_2$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$d_3$</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>$d_4$</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$d_5$</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>$d_6$</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>$d_7$</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>$q$</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>
## Computation of $C_{i,r}$

<table>
<thead>
<tr>
<th></th>
<th>$K_1$</th>
<th>$K_2$</th>
<th>$K_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_1$</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$d_2$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$d_3$</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>$d_4$</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$d_5$</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>$d_6$</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>$d_7$</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>$q$</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$K_1$</th>
<th>$K_2$</th>
<th>$K_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_1 = m_6$</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$d_2 = m_2$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$d_3 = m_7$</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$d_4 = m_2$</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$d_5 = m_8$</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$d_6 = m_7$</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$d_7 = m_3$</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$q = m_8$</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$C_{1,r}$</th>
<th>$C_{2,r}$</th>
<th>$C_{3,r}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_1$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$m_2$</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$m_3$</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>$m_4$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$m_5$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$m_6$</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$m_7$</td>
<td>0</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>$m_8$</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
Computation of Index Term Vectors

\[ k_1 = \frac{(3m_2 + 2m_6 + m_8)}{\sqrt{3^2 + 2^2 + 1^2}} \]

\[ k_2 = \frac{(5m_3 + 3m_7 + 2m_8)}{\sqrt{5 + 3 + 2}} \]

\[ k_3 = \frac{(1m_6 + 5m_7 + 4m_8)}{\sqrt{1 + 5 + 4}} \]

<table>
<thead>
<tr>
<th></th>
<th>(C_{1,r})</th>
<th>(C_{2,r})</th>
<th>(C_{3,r})</th>
</tr>
</thead>
<tbody>
<tr>
<td>(m_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(m_2)</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(m_3)</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>(m_4)</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(m_5)</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(m_6)</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>(m_7)</td>
<td>0</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>(m_8)</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
Computation of Document Vectors

\[ d_1 = 2k_1 + k_3 \]
\[ d_2 = k_1 \]
\[ d_3 = k_2 + 3k_3 \]
\[ d_4 = 2k_1 \]
\[ d_5 = k_1 + 2k_2 + 4k_3 \]
\[ d_6 = 2k_2 + 2k_3 \]
\[ d_7 = 5k_2 \]
\[ q = k_1 + 2k_2 + 3k_3 \]

<table>
<thead>
<tr>
<th></th>
<th>(K_1)</th>
<th>(K_2)</th>
<th>(K_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_1)</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>(d_2)</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(d_3)</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>(d_4)</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>(d_5)</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>(d_6)</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>(d_7)</td>
<td>0</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>(q)</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>
Conclusions

- Model considers correlations among index terms
- Not clear in which situations it is superior to the standard Vector model
- Computation costs are higher
- Model does introduce interesting new ideas
Latent Semantic Indexing
Latent Semantic Indexing

Classic IR might lead to poor retrieval due to:
- unrelated documents might be included in the answer set
- relevant documents that do not contain at least one index term are not retrieved

**Reasoning:** retrieval based on index terms is vague and noisy

The user information need is more related to concepts and ideas than to index terms

A document that shares concepts with another document known to be relevant might be of interest
Latent Semantic Indexing

The key idea is to map documents and queries into a lower dimensional space (i.e., composed of higher level concepts which are in fewer number than the index terms)

Retrieval in this reduced concept space might be superior to retrieval in the space of index terms
Latent Semantic Indexing

**Definitions**

- Let $t$ be the total number of index terms.
- Let $N$ be the number of documents.
- Let $\vec{M} = (M_{ij})$ be a term-document matrix with $t$ rows and $N$ columns.
- To each element of this matrix is assigned a weight $w_{i,j}$ associated with the pair $[k_i, d_j]$.
- The weight $w_{i,j}$ can be based on a *tf-idf* weighting scheme.
Latent Semantic Indexing

The matrix $\vec{M} = (M_{ij})$ can be decomposed into 3 matrices (singular value decomposition) as follows:

$$\vec{M} = \vec{K} \vec{S} \vec{D}^t$$

- $\vec{K}$ is the matrix of eigenvectors derived from $\vec{M} \vec{M}^t$
- $\vec{D}^t$ is the matrix of eigenvectors derived from $\vec{M}^t \vec{M}$
- $\vec{S}$ is an $r \times r$ diagonal matrix of singular values where $r = \min(t, N)$ that is, the rank of $\vec{M}$
Computing an Example

Let \( \vec{M} = (M_{ij}) \) be given by the matrix

\[
\begin{array}{ccc|c}
   & K_1 & K_2 & K_3 & q \cdot d_j \\
\hline
d_1 & 2 & 0 & 1 & 5 \\
d_2 & 1 & 0 & 0 & 1 \\
d_3 & 0 & 1 & 3 & 11 \\
d_4 & 2 & 0 & 0 & 2 \\
d_5 & 1 & 2 & 4 & 17 \\
d_6 & 1 & 2 & 0 & 5 \\
d_7 & 0 & 5 & 0 & 10 \\
q & 1 & 2 & 3 & \\
\end{array}
\]

Compute the matrices \( \vec{K} \), \( \vec{S} \), and \( \vec{D}^t \)

Latent Semantic Indexing

- In the matrix $\vec{S}$, select only the $s$ largest singular values.
- Keep the corresponding columns in $\vec{k}$ and $\vec{D}^t$.
- The resultant matrix is called $\vec{M}_s$ and is given by
  \[ \vec{M}_s = \vec{K}_s \vec{S}_s \vec{D}^t_s \]
  where $s$, $s < r$, is the dimensionality of the concept space.
- The parameter $s$ should be
  - large enough to allow fitting the characteristics of the data.
  - small enough to filter out the non-relevant representational details.
Latent Ranking

- The user query can be modelled as a pseudo-document in the original $\vec{M}$ matrix.
- Assume the query is modelled as the document numbered $0$ in the $\vec{M}$ matrix.
- The matrix $\vec{M}_s^t \vec{M}_s$ quantifies the relationship between any two documents in the reduced concept space.
- The first row of this matrix provides the rank of all the documents with regard to the user query (represented as the document numbered $0$).
Conclusions

Latent semantic indexing provides an interesting conceptualization of the IR problem.

It allows reducing the complexity of the underline representational framework which might be explored, for instance, with the purpose of interfacing with the user.
Neural Network Model
Neural Network Model

Classic IR:
- Terms are used to index documents and queries
- Retrieval is based on index term matching

Motivation:
- Neural networks are known to be good pattern matchers
Neural Network Model

Neural Networks:
- The human brain is composed of billions of neurons
- Each neuron can be viewed as a small processing unit
- A neuron is stimulated by input signals and emits output signals in reaction
- A chain reaction of propagating signals is called a spread activation process
- As a result of spread activation, the brain might command the body to take physical reactions
A neural network is an oversimplified representation of the neuron interconnections in the human brain:

- nodes are processing units
- edges are synaptic connections
- the strength of a propagating signal is modelled by a weight assigned to each edge
- the state of a node is defined by its activation level
- depending on its activation level, a node might issue an output signal
Neural Network for IR

From the work by Wilkinson & Hingston, SIGIR’91

Retrieval Evaluation, Modern Information Retrieval, Addison Wesley, 2006 – p. 49
Neural Network for IR

- Three layers network
- Signals propagate across the network
- First level of propagation:
  - Query terms issue the first signals
  - These signals propagate across the network to reach the document nodes
- Second level of propagation:
  - Document nodes might themselves generate new signals which affect the document term nodes
  - Document term nodes might respond with new signals of their own
Quantifying Signal Propagation

- Normalize signal strength (MAX = 1)
- Query terms emit initial signal equal to 1
- Weight associated with an edge from a query term node $k_i$ to a document term node $k_i$:

$$\overline{w}_{i,q} = \frac{w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,q}^2}}$$

- Weight associated with an edge from a document term node $k_i$ to a document node $d_j$:

$$\overline{w}_{i,j} = \frac{w_{i,j}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2}}$$
Quantifying Signal Propagation

After the first level of signal propagation, the activation level of a document node $d_j$ is given by:

$$
\sum_{i=1}^{t} w_{i,q} \overline{w}_{i,j} = \frac{\sum_{i=1}^{t} w_{i,q} w_{i,j}}{\sqrt{\sum_{i=1}^{t} w_{i,q}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,j}^2}}
$$

which is exactly the ranking of the Vector model

New signals might be exchanged among document term nodes and document nodes in a process analogous to a feedback cycle

A minimum threshold should be enforced to avoid spurious signal generation
Conclusions

- Model provides an interesting formulation of the IR problem
- Model has not been tested extensively
- It is not clear the improvements that the model might provide