



中国科学技术大学

University of Science and  
Technology of China (USTC)

# Content-based Clustered P2P Search Model Depending on Set Distance

Jing Wang

[joycew@mail.ustc.edu.cn](mailto:joycew@mail.ustc.edu.cn)

Network and Information Center

Department of Computer Science and Technology

12-17-2006





# *Agenda*

---

- ◆ Motivations
- ◆ Related Works
- ◆ Document Similarity Definition Based on Set Distance
- ◆ Content-based Clustered Search Model
- ◆ Simulation Experiment
- ◆ Conclusion and Future Works





# Motivation

- ◆ Key issues: index and locate the resources effectively
  - Unstructured P2P System
    - Napster
    - Gnutella
  - Structured P2P System (DHT)
    - Advantages: good scalability
    - Disadvantages: only support exact match
- ◆ My idea → Construct a Search Model Based on Content





# *Agenda*

---

- ◆ Motivations
- ◆ **Related Works**
- ◆ Document Similarity Definition Based on Set Distance
- ◆ Content-based Clustered Search Model
- ◆ Simulation Experiment
- ◆ Conclusion and Future Works





## *Related Works*

- ◆ SETS: Search Enhanced by Topic Segmentation
  - Arrange participating sites in a topic-segmented overlay network
  - A central server: single of bottleneck
- ◆ Locating data in P2P Scientific Collaborations
  - Clustering peers according to their interests
  - Difficult to calculate the similarity between documents
- ◆ Documents Similarity Calculation: Vector Space Model
  - High computational complexity
  - Unsuitable for high dimension sparse matrix
- ◆ **Our solution**
  - Document Similarity Computation: Set distance
  - Architecture of the model: Content-based





# *Agenda*

---

- ◆ Motivations
- ◆ Related Works
- ◆ Document Similarity Definition Based on Set Distance
- ◆ Content-based Clustered Search Model
- ◆ Simulation Experiment
- ◆ Conclusion and Future Works





# Document Similarity Definition Based on Set Distance

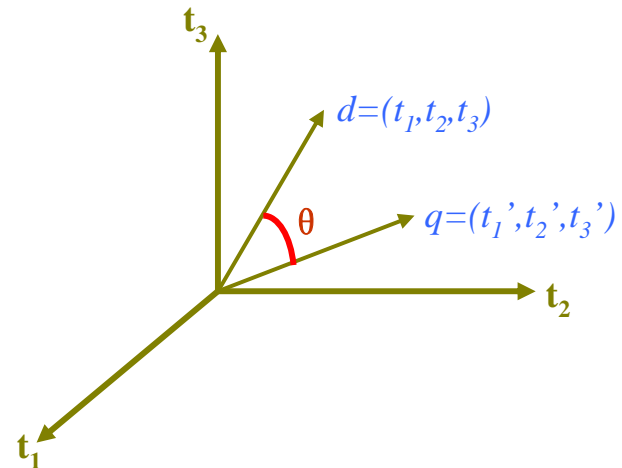
## ◆ Corpus A: Term-by-Document Matrix

- Vocabulary=( $T_1, T_2, \dots, T_n$ )
- Document=( $t_1, t_2, \dots, t_n$ )

$$t_i = \begin{cases} 1, & \text{if } T_i \text{ appeared in document;} \\ 0, & \text{if } T_i \text{ not appeared in document.} \end{cases}$$

TermID \ DocID	$T_1$	$T_2$	.....	$T_n$
$D_1$	0	1	.....	0
$D_2$	1	1	.....	1
.....	.....	.....	.....	.....
$D_m$	1	0	.....	1

## ◆ Similarity Calculation



VSM: Similarity ( $d, q$ ) =  $\cos \theta$

$$= \frac{\sum_{k=1}^3 t_k * t_k'}{\sqrt{(\sum_{k=1}^3 t_k^2)(\sum_{k=1}^3 (t_k')^2)}}$$

Our Solution: Set Distance





# Document Similarity Definition Based on Set Distance

## ◆ Some Definitions

### ■ Set Distance of Corpus A

- $SD(A) = \frac{val_0}{|A| * val_1}$  ,  $val_1$  indicates the number of attributes that valued 1 in all documents  
 $val_0$  indicates the number of attributes that valued 1 for some documents while valued 0 for other documents.

### ■ Document Similarity

- $Sim(d_1, d_2) = 1 / SD(d_1, d_2) = 2val_1 / val_0$

## ◆ An example

- $d_1 = (1, 0, 1, 1, 1, 1)$ ,  $d_2 = (1, 0, 0, 0, 0, 1)$ ,  $d_3 = (1, 0, 1, 1, 0, 0)$

- VSM:  $Sim(d_1, d_2) = 0.6324$ ,  $Sim(d_1, d_3) = 0.7746$

- Our solution:  $Sim(d_1, d_2) = 1.3333$ ,  $Sim(d_1, d_3) = 3.0001$

↓ *mergence (divided by number of terms in vocabulary)*

$$Sim(d_1, d_2) = 0.2667, Sim(d_1, d_3) = 0.6000$$

## ◆ Advantage: reduce the computing complexity in linear time





# *Agenda*

---

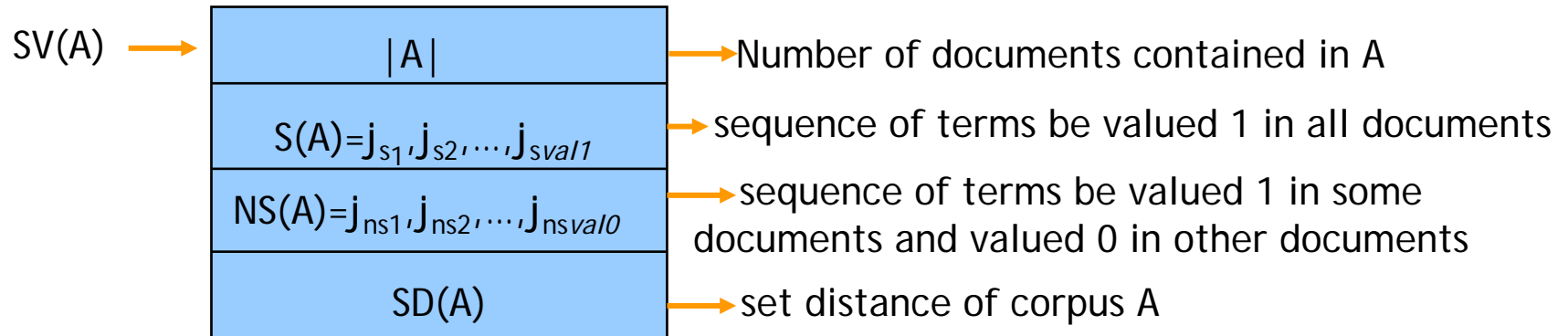
- ◆ Motivations
- ◆ Related Works
- ◆ Document Similarity Definition Based on Set Distance
- ◆ **Content-based Clustered Search Model**
- ◆ Simulation Experiment
- ◆ Conclusion and Future Works





# Content-based Clustered Search Model

## ◆ A Definition: sparse vector (SV) of corpus A



## ◆ Feature of Additivity: $SV(A \cup B) = SV(A) + SV(B)$





# Content-based Clustered Search Model

## ◆ Content-based Search Model Connected by Superpeers

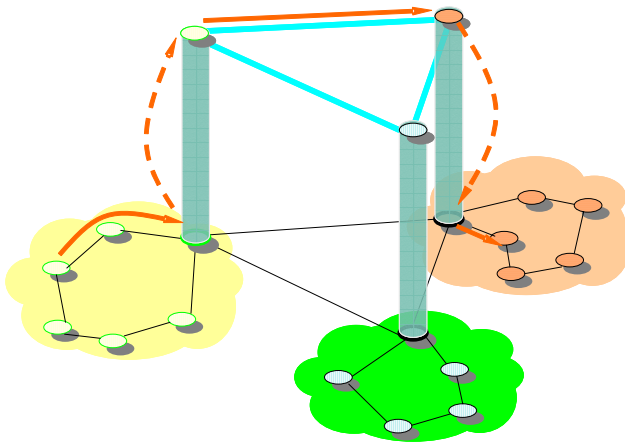


Fig.1. Content-based Search Mode

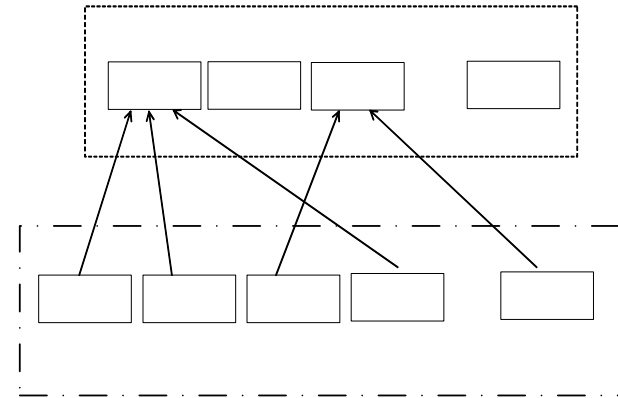
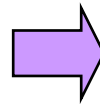


Fig.2. Structure of the model based on Feature Vector

## ◆ Construction and Implementation





# *Agenda*

---

- ◆ Motivations
- ◆ Related Works
- ◆ Document Similarity Definition Based on Set Distance
- ◆ Content-based Clustered Search Model
- ◆ **Simulation Experiment**
- ◆ Conclusion and Future Works





# Simulation Experiment

## ◆ Performance parameters

{ query result → recall  
{ query efficiency → query time & visited nodes

## ◆ Experiment parameters

Parameters	Definition	Default
$Number_{peer}$	Number of peers in the system	1000
$Number_{words}$	Number of keywords in the vocabulary every superpeer maintains	1000
$Key_{peer}$	Number of keywords in the inverted table of each peer	150
$TTL$	Time-to-Live	6
$Threshold_{cluster}$	Threshold of cluster	1
$Similarity_{docume}$	Threshold of every two documents	1
$K$	Number of peers that superpeer choose to forward the query	6





# Simulation Experiment

## ◆ Query Effect

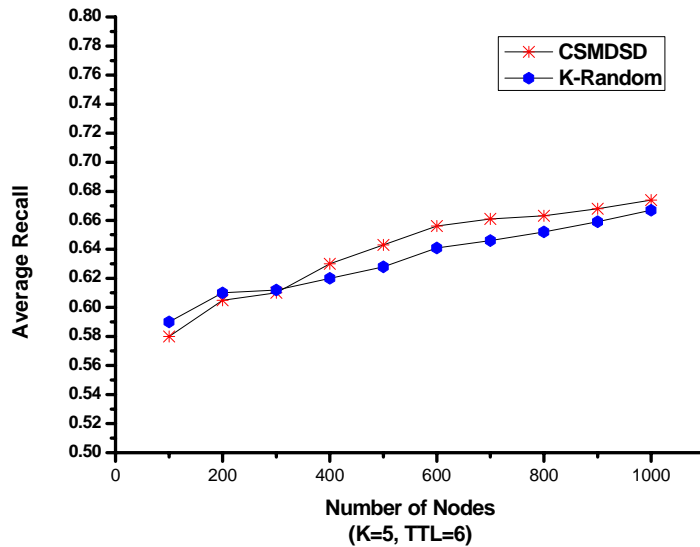


Fig.3. Comparison of average recall between CSMDSD and K-Random

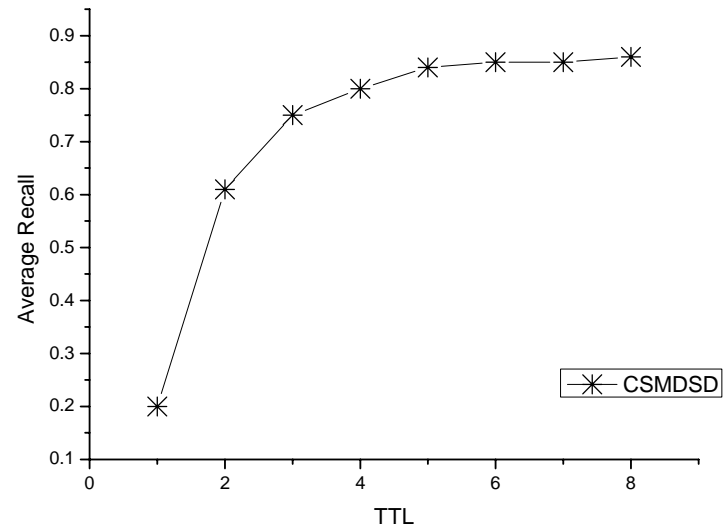
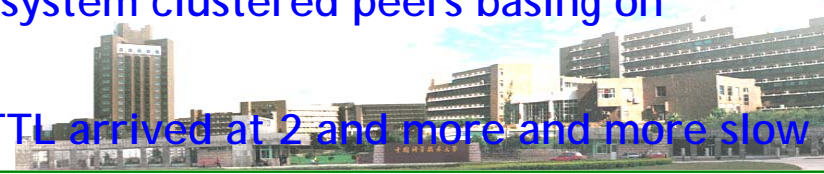


Fig.4. Effects of TTL on the average recall

1. With the increase of number of peers, our system clustered peers basing on content and the recall is improved;
2. The increase of recall is the biggest with TTL arrived at 2 and more and more slow





# Simulation Experiment

## ◆ Query Efficiency

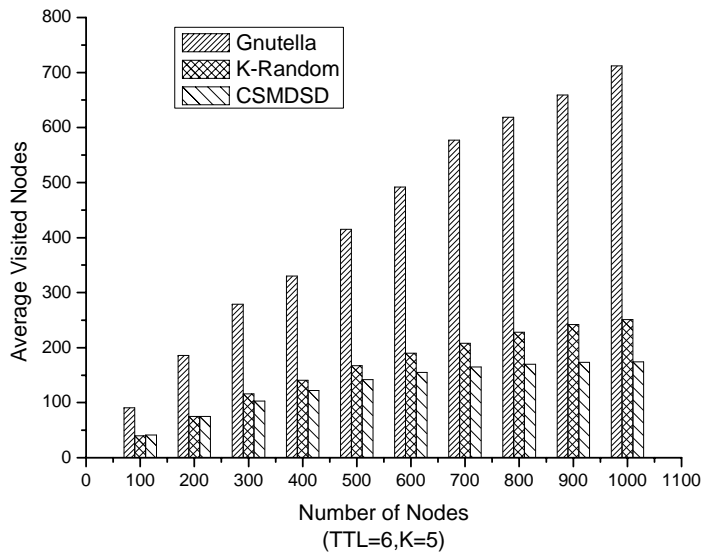


Fig.5. Average visited nodes

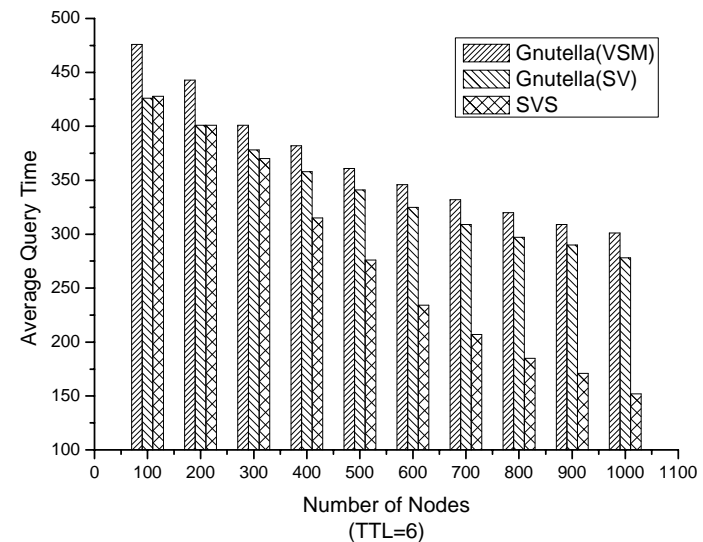
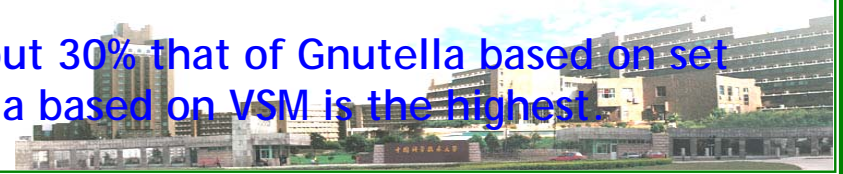


Fig.6. Average query time

1. The average visited peers of K-random and CSMDSD is lower than half of that of Gnutella and the increase of visited nodes of K-random is faster than CSMDSD;
2. The average query time of CSMDSD is about 30% that of Gnutella based on set distance, while the query time of Gnutella based on VSM is the highest.





# *Agenda*

---

- ◆ Motivations
- ◆ Related Works
- ◆ Document Similarity Definition Based on Set Distance
- ◆ Content-based Clustered Search Model
- ◆ Simulation Experiment
- ◆ Conclusion and Future Works





# *Conclusion and Future Works*

## ◆ Conclusions

- Compresses data of document vector
- Reduces calculations and improves the query efficiency
- Clusters peers based on content

## ◆ Future works

- The system is based on an assumption, that the two sets have no intersection, which means that the duplicate management isn't considered and need to be improved.
- Be implemented in application





*Thanks!*

---

Questions and comments?

