Efficient Token-based Web-snippet Clustering

Li Jianchao
Outline

- Motivations
- System Processing Flowchart
- Basic Concepts
- Algorithms
- Result
Motivations

- Each cluster can be identified by some common words
- Clustering model based on VSM should be replaced to capture sentence structures
- Character-based clustering may be an alternative for word-based clustering, and it’s very useful for oriental languages
- Soft Clustering is better for web clustering
System Processing Flowchart

- Search Engine
- Web Titles and Snippets
- Directed Probability Graph Construction
- Visualization
  - Cluster Packing
  - Phrase1
  - Phrase2
  - Phrase3
- Frequent Phrase Identification
Basic Concepts

- **Token:**
  1. An occident language word
  2. Sequential digital number
  3. Other character such as oriental characters except sentence punctuations
  4. Repeated units above
- **DPG:**
  Directed Probability Graph
- **Phrase:**
  Meaningful sequential characters
Algorithms

- Two-token Frequent Phrase Identification
- Phrase Extending
- Cluster Packing
Two-token Frequent Phrase Identification

• Concepts:
  – Global Support: the percentage of documents that appear the phrase in all documents
  – TF-IDF: term frequency * inverse document frequency
  – Ratio: the ratio of an edge appears for each node

• Algorithm
  – Go through all edges for calculating limitation
  – Go through again for frequent two-token phrases than satisfy the limitation
Phrase Extending & Clustering

- Coherence $=1 - \text{abs}(T_1.st - T_2.st)/T_1.st$
  where $T_1, T_2$ are two-token phrases and "st" denotes the support value of $T_1, T_2$

- Algorithm
  - For all two-token frequent phrases
    - if two phrase of them $T_1, T_2$ are joint and coherent with each other, combine them into one, delete the later one and continue for further extending
    - else if they are joint but not coherent
      - if $T_1.st > T_2.st$ & $T_1$ cover $T_2$, pre-merge $T_1$ to $T_2$
      - If $T_1.st < T_2.st$ & $T_2$ cover $T_1$, post-merge $T_2$ to $T_1$
Cluster Packing

- Pack similar clusters into one if they have large overlap or synonymous labels
- Determine whether two label are synonyms based on Chinese Synonym Dictionary
- Labels are concatenated using comma as the final cluster label
Result Analysis

- Generate clusters with little overlaps
- Extract meaningful labels with enough records
- Less time complexity and more efficient
THE END

Thank you!