Constructing Hierarchical Relationships of Domain-Specific Terminology in Chinese: Post-print

Authors: Zhu Hui, Yang Jianlin, Wang Hao

Date: 2017-10-11T00:00:00+00:00

Abstract

Purpose: This study explores methods for extracting hierarchical relationships of terms from unstructured Chinese text.

Methodology: Literature in the digital library domain was retrieved from CNKI. A semantic hierarchical structure of terms was constructed through term extraction, term vector space model construction, BIRCH algorithm clustering, and cluster label determination.

Results: A hierarchical structure of terms in the digital library domain was constructed and validated. The clustering accuracy reached 80.88%, while the class label extraction accuracy achieved 89.71%.

Limitations: Validation of the construction effectiveness was conducted through random sampling and only empirically compared with one alternative construction method.

Conclusion: Applying BIRCH algorithm clustering to construct term hierarchical structures demonstrates significant advantages compared to K-means clustering, offering high execution efficiency and clustering effectiveness.

Full Text

Research on Constructing Taxonomic Relations of Domain-Specific Chinese Terminology*

Zhu Hui, Yang Jianlin, Wang Hao

(School of Information Management, Nanjing University, Nanjing 210023, China)
(Jiangsu Key Laboratory of Data Engineering and Knowledge Services, Nanjing 210023, China)

Abstract

[**Objective**] This study explores how to extract terminological taxonomic relations from Chinese unstructured text. [**Methods**] We collected documents from the digital library discipline via CNKI and constructed a semantic hierarchy through terminology extraction, vector space model construction, BIRCH algorithm clustering, and cluster label determination. [**Results**] The taxonomic structure for digital library terminology was successfully constructed and validated, achieving a clustering accuracy of 80.88% and a cluster label extraction accuracy of 89.71%. [**Limitations**] The validation was conducted through random sampling and compared with only one alternative method. [**Conclusions**] The BIRCH algorithm demonstrates clear advantages over K-means clustering for constructing terminological hierarchies, offering higher execution efficiency and clustering effectiveness.

Keywords: Terminology; Taxonomic Relation; Ontology; Ontology Learning; Clustering
Classification Number: TP391

Taxonomic relations among domain terms constitute a crucial component of domain knowledge ontologies. By organizing terms hierarchically according to categories, these relations enable domain knowledge search, reuse, and deeper understanding. Some scholars even argue that ontologies themselves represent hierarchical structures among concepts with inclusion relationships [1-2]. Manual construction of term hierarchies is time-consuming, labor-intensive, and constrained by domain experts' background knowledge, lacking objectivity and consistency. Consequently, employing automated knowledge acquisition methods to build term hierarchies has emerged as a new research direction.

One common approach for acquiring term taxonomic relations is based on Harris' distributional hypothesis [3], which states that if two terms share similar contextual environments, they are semantically similar [4]. Researchers have validated this hypothesis and confirmed its effectiveness [5]. Building upon Harris' hypothesis, clustering methods can be introduced to construct term hierarchies.

This paper proposes a concrete method for building domain term taxonomic relations from Chinese unstructured text by establishing a term vector space model, introducing the BIRCH algorithm and term co-occurrence theory into domain ontology construction, and optimizing clustering results through vector space model improvements. The BIRCH algorithm, designed for large-scale data clustering, has been applied in text clustering and large-scale network data clustering, but not previously in term hierarchy construction. This study explores its application in building term hierarchies and compares it with other clustering methods.

3. Term Taxonomic Relation Acquisition Based on BIRCH Clustering

This section focuses on the methodology and process of acquiring term taxonomic relations from unstructured text using BIRCH clustering, analyzing the construction and optimization of the term vector space model.

3.1 Term Extraction Researchers are direct participants and witnesses of dynamic terminology evolution in a discipline, and their scholarly publications document this developmental process. Since keywords in these documents distill research content, domain terms can be extracted from publication keywords. However, author-assigned keywords suffer from arbitrariness, inconsistency, and errors, necessitating standardization to ensure unique representation of each concept.

Domain terms must possess recognized status within the field. This study uses the frequency of occurrence of a keyword across all documents (Nk) as a filtering criterion. A keyword is considered a domain term if:

$$N_k \geq C$$

where C represents the frequency threshold.

3.2 Construction of Term Vector Space Model Describing terms using documents as features forms the foundation for subsequent clustering. Using the term set as a dictionary, the Chinese word segmentation tool NLPIR establishes semantic associations between documents and terms [19] to construct a document-term frequency matrix. After TF-IDF weighting calculation, we obtain a term-document weight matrix.

In this term-document vector space model, measuring term similarity relies on term co-occurrence in documents. In shorter unstructured documents, limited term quantities result in sparse co-occurrence relationships and a sparse term-document matrix. Mining hierarchical relations from such sparse matrices may yield unsatisfactory results. How can we increase term co-occurrence to densify the matrix?

In the term-document model, documents mediate term co-occurrence: if terms T1 and T2 both associate with document Di, they co-occur. Since documents comprise many words, we can consider T1 as associated with all wi words in Di, expanding one term-document association into wi term-word associations. Similarly, T2 also associates with all wi words in Di, creating term co-occurrence mediated by words. This mediation transformation significantly changes co-occurrence patterns: terms that originally co-occurred maintain and strengthen their relationships, while previously unrelated terms become associated if their respective documents share common words [20].

Using NLPIR with the term set as a user dictionary, we segment unstructured documents, select nouns, and remove stopwords and low-frequency words to obtain the required vocabulary. Terms find associated words through their linked documents, generating <term, word, co-occurrence frequency> triples. We further introduce the Ochiia coefficient to measure association strength, forming <term, word, association coefficient> triples to construct a term-word weight matrix.

3.3 Two-Step Clustering Based on the term vector space model, we employ a two-step clustering approach: first using BIRCH for pre-clustering to obtain "coarse" results, then applying hierarchical clustering to derive the term hierarchy. Categories not meeting termination conditions undergo repeated two-step clustering, making the entire process a multi-level two-step clustering, as illustrated in [Figure 1: see original paper].

The BIRCH algorithm uses Clustering Features (CF) and CF-trees. A node j in the CF-tree represents class j, denoted as CFj, comprising three components: {N_j, S_{Aj}, S_{Aj}}, where N_j is the number of terms, S_{Aj} is the linear sum of N_j terms, and S_{Aj} is the square sum.

For example, if node CF1 contains three data points: (1,2), (3,4), (5,6), then CF1 = {3, (1+3+5, 2+4+6), (1^2+3^2+5^2, 2^2+4^2+6^2)} = {3, (9,12), (35,56)}.

For a new class <j, s> formed by merging classes j and s: {N_j+N_s, S_{Aj}+S_{As}, S_{Aj}+S_{As}}.

The BIRCH algorithm proceeds as follows:
1) Treat all terms as one class, calculate CF, and create the root node.
2) Read a term, calculate its log-likelihood distance from intermediate nodes starting from the root, and traverse down the path with minimum distance until reaching a leaf node.
3) Calculate distances between the term and all leaf nodes in the subtree. If the minimum distance is below threshold T, absorb the term. If the leaf node exceeds capacity, split it into an intermediate node and recalculate CF for leaf and parent nodes. Otherwise, create a new leaf node and recalculate CF for all parent nodes.
4) Check if the number of leaf nodes reaches the maximum. If all terms are processed, end clustering; otherwise, adjust threshold T and rebuild a smaller CF-tree.

The two-step clustering automatically determines cluster numbers through two phases:

(1) Phase One: Uses Bayesian Information Criterion (BIC) as the decision standard. Assuming J clusters:

$$BIC(J) = -2\sum_{j=1}^{J} \xi_j + m \log N$$

where the first term reflects the total log-likelihood (within-class variation) and the second term penalizes model complexity. When all samples merge into one class, the first term is maximized and the second minimized. As pre-clustering numbers increase, the first term decreases while the second increases, with total value decreasing until reaching an inflection point J, which provides a "coarse" estimate.

(2) Phase Two: Refines the coarse estimate J using the metric:

$$R^2(J) = \frac{\min_{j \neq s}(C_{js})}{\min_{j \neq s}(C_{js})}$$

where $\min(C_{js})$ is the minimum inter-class log-likelihood distance. $R^2(J)$ reflects changes in minimum inter-class variation during hierarchical merging—larger values indicate less appropriateness of merging J+1 classes into J. By calculating $R^2(J-1)$, $R^2(J-2)$ through $R^2(2)$, we identify the maximum and second-maximum values. If the maximum exceeds the second-maximum by 1.15 times, its corresponding J becomes the final cluster number; otherwise, the larger of the two J values is selected.

3.4 Cluster Label Determination Establishing the term hierarchy simultaneously involves determining cluster labels. For each category at every level, we calculate each term' s comprehensive semantic similarity within the class and select the term with maximum similarity as the cluster label [11].

Given terms $T_i = (w_{i1}, w_{i2}, \cdots, w_{im})$ and $T_j = (w_{j1}, w_{j2}, \cdots, w_{jm})$, their semantic similarity is:

$$Sim(T_i, T_j) = \frac{\sum_{k=1}^{m} w_{ik} w_{jk}}{\sqrt{\sum_{k=1}^{m} w_{ik}^2} \sqrt{\sum_{k=1}^{m} w_{jk}^2}}$$

Comprehensive semantic similarity of term T_i is the sum of its similarities with all other terms in the class:

$$SumSim(T_i) = \sum_{j=1, j \neq i}^{n} Sim(T_i, T_j)$$

The term with maximum comprehensive semantic similarity represents the broadest semantic content and serves as the cluster label.

4. Experimental Results and Analysis

Using digital library domain journal papers as our analysis object, we performed clustering based on term-word semantic associations and validated the constructed term hierarchy.

4.1 Data Preprocessing We retrieved 7,746 papers published between 1996-2011 from CNKI' s core journal database using "digital library" as the subject term, extracting titles, abstracts, and keywords as unstructured documents. Term extraction yielded 911 terms. Using these as a user dictionary for NLPIR segmentation produced 50,992 term-document associations. When using words as co-occurrence mediators, segmentation and filtering yielded 2,168 words and 105,477 term-word semantic associations—significantly more than term-document associations, creating a denser vector space.

4.2 Determination of Cluster Numbers Our two-step clustering automatically determines cluster numbers. The scheme involves: domain experts specifying range limits for each level, then the algorithm selecting optimal numbers within those ranges. Let n denote the number of terms in a class, MaxNum the maximum terms allowed before stopping clustering, and Ceil(X) the smallest integer $\geq$ X. Based on domain characteristics, we set: first-level range 10-15, second-level 5-10, and subsequent levels dependent on class size: if n $\geq$ 5$\times$MaxNum, the range is 5-Ceil(n/MaxNum), otherwise Ceil(n/MaxNum)-5.

Since optimal MaxNum is unknown, we tested values {5,10,15,20}. Results appear in . A good hierarchy requires reasonable depth, width, and intra-class node distribution. Based on domain characteristics and result observation, we selected MaxNum=10. Table 2 shows first-level cluster data.

4.3 Analysis of Clustering Results Clustering analysis is unsupervised; different parameters and designs yield different results. Without unified evaluation standards, we validated results through domain expert review. We randomly sampled 10 parent classes with their subclasses to examine clustering effectiveness and the reasonableness of label extraction.

For each parent class and its subclasses, we assessed: (1) relationships among subclass labels—strong associations among most labels indicated good clustering; (2) relationships between subclass and parent labels—strong associations indicated reasonable label extraction. Data appears in .

Results show:
(1) Clustering Effectiveness: Most sampled classes exhibited strong intra-member relationships, with clustering accuracy $\geq$ 66.67% and an average of 80.88%, demonstrating effective clustering of sparse data. For example, class "C3_{Knowledge} Service" contained five subclasses: "C3_1_{Ontology}," "C3_2_{Knowledge} Network," "C3_3_{Semantic} Grid," "C3_5_{Integrated} Service," and "C3_4_{3G}." The first four showed strong interrelationships, while "C3_4_{3G}" lacked clear associations, yielding 80% accuracy.

(2) Label Extraction: Most class labels were reasonably extracted, associating with many members. Label extraction accuracy $\geq$ 66.67%, averaging 89.71%. For "C3_{Knowledge} Service," four of five subclass labels strongly

associated with the parent label, giving 80% accuracy.

4.4 Comparison with K-means Clustering We compared BIRCH with K-means clustering for hierarchy construction, with results in .

Comparison reveals:
(1) **Advantage in cluster number determination:** K-means requires specifying cluster numbers, demanding time-consuming trial-and-error for reasonable results. BIRCH automatically determines numbers within specified ranges using relevant metrics.

(2) **Suitability for sparse data:** K-means produced many single-term clusters that could have been merged, while BIRCH exhibited this phenomenon less frequently.

(3) **More reasonable hierarchy dimensions:** K-means generated twice as many total clusters as BIRCH, with maximum depth reaching 18—failing to objectively reflect actual term hierarchies.

(4) **Higher effectiveness:** K-means achieved 70.39% average clustering accuracy and 55.59% label extraction accuracy, both lower than BIRCH' s 80.88% and 89.71%.

Conclusion

This paper proposes a method for acquiring term taxonomic relations from domain unstructured text through term extraction, vector space model construction, BIRCH clustering, and label determination. By replacing term-document vector space with term-word space, we increased data density, providing a solid foundation for BIRCH clustering. Compared with related methods, BIRCH offers clear advantages: suitability for large datasets, outlier/noise diagnosis, and automatic cluster number determination. Using the digital library domain, we demonstrated the method' s feasibility and effectiveness. Limitations include validation based solely on random sampling and comparison with only one alternative method. Future work will explore different machine learning methods for (semi-)automated term hierarchy acquisition and investigate more effective strategies.

References

[1] Gruber T R. A Translation Approach to Portable Ontology Specifications [J]. Knowledge Acquisition, 1993, 5(2): 199-220.

[2] Rios-Alvarado A B, Lopez-Arevalo I, Sosa-Sosa V J. Learning Concept Hierarchies from Textual Resources for Ontologies Construction [J]. Expert Systems with Applications, 2013, 40(15): 5907-5915.

[3] Wen Chun, Shi Zhaoxiang, Zhang Xiao. A Survey on Ontology Concept Hierarchy Acquisition [J]. Computer Applications and Software, 2010, 27(9):

103-107.

[4] Harries Z S. Mathematical Structures of Language [M]. New York: Wiley, 1968.

[5] Miller G A, Charles W. Contextual Correlates of Semantic Similarity [J]. Language and Cognitive Processes, 1991, 6(1): 1-28.

[6] Sun C, Zhao M, Long Y J. Learning Concepts and Taxonomic Relations by Metric Learning for Regression Communications in Statistics-Theory and Methods, 2014, 43(14): 2938-2950.

[7] Hu F H, Shao Z Q, Ruan T. Self-Supervised Chinese Ontology Learning from Online Encyclopedias [J]. The Scientific World Journal, 2014: Article ID 848631.

[8] Colace F, De Santo M, Greco L, et al. Terminological Ontology Learning and Population Using Latent Dirichlet Allocation [J]. Journal of Visual Languages and Computing, 2014, 25(6): 818-826.

[9] Meijer K, Frasincar F, Hogenboom F. A Semantic Approach for Extracting Domain Taxonomies from Text [J]. Decision Support Systems, 2014, 62: 78-93.

[10] De Knijff J, Frasincar F, Hogenboom F. Domain Taxonomy Learning from Text: The Subsumption Method Versus Hierarchical Clustering[J]. Data & Knowledge Engineering, 2013, 83: 54-69.

[11] Ji Peipei, Yan Xiaoyan, Cen Yonghua, et al. Research of Term Semantic Hierarchy Induction Domain-specific Chinese Text Information Processing [J]. New Technology of Library and Information Service, 2010(9): 37-41.

[12] Lin Yuan, Chen Zhibo, Sun Qiao. Computer Domain Term Automatic Extraction and Hierarchical Structure Building [J]. Computer Engineering, 2011, 37(2): 172-174.

[13] Peng Cheng, Ji Peipei. Research of Term Semantic Hierarchy Correlations Based on Deterministic Annealing [J]. Application Research of Computers, 2011, 28(9): 3235-3238.

[14] Gu Jun, Zhu Ziyang. Study on Ontology Hierarchy Relation Induction on Clustering Algorithm [J]. New Technology of Library and Information Service, 2011(12): 46-51.

[15] Han Hongqi, Xu Shuo, Gui Jie, et al. Term Hierarchical Relation Extraction Method Based on Morphology Rule Template [J]. Journal of the China Society for Scientific and Technical Information, 2013, 32(7): 708-715.

[16] Tu Ding, Chen Ling, Chen Gencai, et al. Multi-way Hierarchical Clustering Based Concept Taxonomy Construction for Product Reviews [J]. Journal of Computer Research and Development, 2013, 50(S): 208-215.

[17] Li Shuqing. Research on Automatic Construction of Domain Ontology in Library and Information Science Based on Weighted Co-occurrence of Citation

Keywords [J]. Journal of the China Society for Scientific and Technical Information, 2012, 31(4): 371-380.

[18] Zhang T, Ramakrishnan R, Livny M. BIRCH: A New Data Clustering Algorithm and Its Applications [J]. Data Mining and Knowledge Discovery, 1997, 1(2): 141-182.

[19] NLPIR [EB/OL]. [2014-06-03]. http://ictclas.nlpir.org/docs.

[20] Wang Hao, Su Xinning, Zhu Hui. Study on Hierarchy Structure Generation of Chinese Medical Terminology [J]. Journal of the China Society for Scientific and Technical Information, 2014, 33(6): 594-604.

Author Contributions

Zhu Hui: Conceived research ideas, designed methodology, conducted experiments, drafted and revised the manuscript.
Yang Jianlin: Conducted literature review and manuscript revision.
Wang Hao: Collected and cleaned data.

Received: 2015-06-19
Revised: 2015-09-14

Note: Figure translations are in progress. See original paper for figures.