Research on Product Domain Ontology Construction Based on Concept Lattice Theory (Postprint)

Authors: Lu Jiaying, Yuan Qinjian, Huang Qi, QIAN Yunjie

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

Abstract

【Purpose】Based on formal concept analysis theory, this study explores and proposes a method for constructing product domain ontologies by reusing product classifications, improving upon previous methods for constructing product domain ontologies through classification reuse. 【Method】Classes, attributes, and relationships are extracted from product classifications, their structure is reasonably adjusted, formal context and concept lattice are generated based on formal concept analysis theory, and product domain ontologies are automatically generated from this foundation. 【Results】The proposed method eliminates the limitation of unclear hierarchies inherent in product classifications, can resolve redundancy issues among attributes defined in product classifications, enhances ontology reusability and shareability, and presents existing and potential entities and relationships in a visual form. 【Limitations】Only a fragment of medical product construction is selected to illustrate the specific implementation of the method, and only the applicability of the method to the eCl@ss product classification system is discussed. 【Conclusion】Before constructing product domain ontologies based on formal concept analysis, it is necessary to clearly define the domain and scope of the ontology, organize and categorize classes according to scientific classification principles, construct an attribute dictionary and define object properties during ontology construction, and automatically generate the ontology through the generation of formal context and concept lattice.

Full Text

Research on Product Domain Ontology Construction Based on Concept Lattice Theory

Lu Jiaying[1], **Yuan Qinjian**[1], **Huang Qi**[2], **Qian Yunjie**[1]
[1](School of Information Management, Nanjing University, Nanjing 210093,

China)
[2](State Information Resources Management Research Base, Nanjing University, Nanjing 210093, China)

Abstract

[**Objective**] This study proposes a method for constructing complex product classification and product domain ontology based on concept lattice theory, aiming to improve existing product classification and domain ontology construction methodologies. [**Methods**] The method extracts class-attribute relationships from product classifications, adjusts the structure appropriately, generates formal contexts and concept lattices based on concept lattice theory, and automatically constructs product domain ontology. [**Results**] Our approach improves the hierarchical structure of product classifications, eliminates attribute redundancy in product classification definitions, enhances ontology reusability and sharability, and visually presents existing and potential entity relationships. [**Limitations**] The study only demonstrates the implementation of the proposed method using a segment of medical product ontology and explores its flexibility for the eCl@ss product classification system. [**Conclusions**] When constructing product domain ontology based on concept lattice, it is essential to clearly define the ontology' s domain and scope, systematically organize categories according to scientific classification principles, build a property dictionary, define object properties, and automatically generate the ontology through formal contexts and concept lattices.

Keywords: Concept Lattice; Domain Ontology; Ontology Construction; Product Ontology

1. Introduction

Concept lattices, also known as Galois lattices, were first proposed by Wille in 1982 [1] and represent the central data structure in formal concept analysis. Philosophically, a concept comprises two fundamental components: extension and intension. Building upon this notion, concept lattices formalize philosophical concepts, where each node represents a formal concept containing two elements: extension (the set of all instances belonging to the concept) and intension (the set of attributes shared by all instances) [2]. Additionally, concept lattice is a lattice structure derived from a partial order relation that explicitly displays generalization and specialization relationships between concepts. Formal concepts and their hierarchical relationships collectively constitute a concept lattice, which is typically visualized using a Hasse diagram—a graphical representation of partial order relations that uses directed edges to reflect inclusion relationships.

Overall, the concept lattice model serves as a powerful tool for data analysis and knowledge discovery, with current applications spanning ontology engineering, data mining, digital library development, and artificial intelligence [3]. Early ontology construction methods typically relied on manual development by domain experts, which was not only costly and time-consuming but also suffered from poor scalability. Moreover, expert-driven conceptual hierarchies often introduced redundancy [4]. Concept lattice generation fundamentally employs an instance-attribute-based approach that can automatically map and generate clear ontology concept hierarchies, providing a foundation for ontology construction. This method supplements and improves upon formal ontology development approaches while avoiding conceptual redundancy. Consequently, ontology construction based on concept lattices exhibits strong methodological rigor and formal semantic completeness, supports intelligent processing of domain texts, and enhances ontology sharability and reusability.

Given concept lattices' advantages in processing conceptual data, an increasing number of scholars have applied them to domain ontology construction. These studies can be divided into two categories: (1) methods generating concept lattices from unstructured data for domain ontology construction, and (2) methods generating concept lattices from structured data for domain ontology construction.

The first category typically employs natural language processing techniques to handle unstructured text (such as web information and documents), extract conceptual hierarchies, and build concept lattices for ontology generation. Haav [5] proposed a semi-automatic ontology design method using formal concept analysis, extracting formal contexts from domain texts through natural language processing, directly mapping formal concepts to ontology concepts, and applying concept lattice reduction to derive initial ontologies. Building on Haav' s work, Cimiano et al. [6] mapped formal concept analysis attributes to ontology concepts, enabling concept lattice construction through natural language analysis for domain ontology generation. Similarly, Liu and Hu [7] utilized CNKI full-text journal articles as data sources, combining association rule mining with formal concept analysis to identify hierarchical relationships between concepts for information science ontology construction.

The second category relies on structured data sources such as relational databases, taxonomies, and classification systems. Nanda et al. [8] developed the PFODM methodology for product metadata ontology, extracting concepts and attributes from material specifications, generating concept hierarchies through formal concept analysis, and constructing domain ontologies. Ouyang et al. [9] proposed an ontology learning method from relational databases, obtaining formal contexts composed of attributes and domain concepts to generate concept lattices and subsequently domain ontologies. Additionally, Teng and Bi [10] used structured heterogeneous resources as a foundation, integrated domain texts, and merged concept lattices from multiple sources to build unified domain ontologies.

Analysis of existing research reveals that scholars have introduced concept lattices into domain ontology construction, yielding valuable results. These methods leverage concept lattices to extract structured information from both unstructured and structured data sources, establishing conceptual hierarchies as the basis for ontology construction. While recent studies have extensively explored concept lattice generation from unstructured data using natural language processing and machine learning, research on concept lattice construction from structured data has primarily focused on relational databases and traditional knowledge organization systems (e.g., taxonomies and classification schemes). For product domain ontology construction, classification systems serve as crucial structured resources that can guide the development process [11]. However, previous ontology construction under classification system guidance relied on expert judgment for conceptual hierarchy generation, creating an opportunity to integrate concept lattice theory into product classification-based ontology development.

Therefore, this study employs concept lattice theory as a foundational methodology to explore how product classifications can be reused for product ontology construction, proposing a more robust product domain ontology construction method that facilitates effective organization and sharing of product and service information among enterprises.

3. Product Domain Ontology Construction Process Based on Concept Lattice Theory

Integrating the principles of the Seven-Step Method [12] and the 骨架法 (Framework Method) [13], this study designs a product domain ontology construction process as illustrated in [Figure 1: see original paper].

[Figure 1: see original paper] Domain Product Ontology Construction Process

The process comprises two main phases: product classification preprocessing (including class/attribute extraction and relationship definition) and concept lattice construction and ontology prototype generation (including formal context generation, concept lattice construction, and ontology formalization). Specifically, the methodology follows seven sequential steps:

(1) **Product Classification Preprocessing**: Extract classes and hierarchical relationships from the product classification, clarify generalization, specialization, part-whole, and aggregation relationships between product categories, and categorize product attributes as data properties or object properties in the ontology. Define object properties to establish relationships between classes and attributes.

(2) **Formalization of Class-Attribute Relationships**: Transform the defined class-attribute relationships into concept-attribute relational tables to create multi-valued formal contexts, represented as quadruples (G, M, W, I), where G is the object set, M is the multi-valued attribute set, W is the attribute value domain, and I is a ternary relation (I G × M × W).

(3) **Conversion to Single-Valued Formal Context**: Since formal concept analysis requires single-valued contexts, convert multi-valued formal contexts to single-valued ones, represented as triples (G, M, I), where G is the object set, M is the single-valued attribute set, and I is a binary relation (I G × M).

(4) **Concept Lattice Generation**: Utilize the OWLFCAView Protégé plugin [14] to automatically convert single-valued formal contexts into concept lattices required for ontology construction.

(5) **Ontology Model Generation**: The OWLFCAView plugin transforms concept lattices into ontology conceptual models, yielding initial ontology prototypes.

(6) **Expert Review and Refinement**: Ontology engineers or domain experts evaluate and iteratively refine the conceptual model until the prototype meets requirements.

(7) **Formalization and Visualization**: Use Protégé to visually model the refined ontology and generate the final domain product ontology.

---

4. Case Study: Product Domain Ontology Construction Based on Concept Lattice

4.1 Product Classification Preprocessing This study employs the eCl@ss product classification as the data source for ontology construction. eCl@ss is a German product and service classification standard and one of the most widely adopted standards for e-commerce environments [15]. The eCl@ss system comprises 30 major categories covering approximately 38,000 product classes and 16,000 product attributes, encompassing virtually all currently used products and services. Specifically, this research utilizes the "Pharmaceutical and Medical Technology" category (Class 34) from eCl@ss version 8.1 as the foundation for ontology construction.

Compared to other product classification systems, eCl@ss offers comprehensive attribute definitions and strong scalability, though its four-level hierarchy limits scientific rationality. Therefore, during class organization, this study transcends eCl@ss's four-level constraint and regroups categories according to scientific classification principles while preserving the original classification codes to ensure compatibility.

The overall adjustment approach for the 52 second-level classes under "Pharmaceutical and Medical Technology" is illustrated in [Figure 3: see original paper]. The restructuring establishes five main categories: Medicine and Active Substance, Medical Equipment, Medical Auxiliary Supplies, Parts, and Accessories. The 52 original second-level classes are merged accordingly, with "parts" and "accessories" classes extracted to form separate Parts and Accessories categories.

Property Definition: In OWL, properties are categorized as data properties (representing relationships between instances and literal values) and object properties (representing relationships between instances) [16]. eCl@ss defines attributes and values for product/service information, providing basic properties (BSP) and specific properties (SSP). From an ontology perspective, eCl@ss attributes are exclusively data properties, which can be directly reused. For example, under Class 34-47 "Medical Auxiliary Supplies," the subclass 34-47-01-01 "Tongue Depressor" includes manufacturer name, product name, product version, and material composition attributes, all reusable as data properties in OWL. Data types should be explicitly defined (boolean, float, int, string) during property creation.

Object Property Definition: Object properties reflect inter-instance relationships. eCl@ss defines several relationship types: class-attribute, part-whole, and accessory-device relationships. This study defines three object properties as shown in Table 1.

Table 1: Object Property Definitions and Semantics

Relationship	Object Property	Semantics
Class-attribute	has_property	A class possesses a specific attribute
Part-whole	is_part_of	A class is a component of another class
Accessory-device	is_accessory_of	A class is an accessory of another class

For instance, "manufacturer name"and "product name"are attributes of "Tongue Depressor," linked via has_property. "Light source" is a part of "Examination Instrument," connected through is_part_of. "Pressure plate" is an accessory of "Small Medical Equipment," related via is_accessory_of.

4.2 Formal Context Generation and Concept Lattice Construction

In eCl@ss, product and service attributes intersect across categories. To clearly illustrate class-attribute relationships and avoid redundancy, this study establishes separate property dictionaries for attributes, parts, and accessories, formalized using axioms like "has_property some values from Property." This axiom, interpreted as "each class must have at least one specified attribute from the Property category," forms the basis for generating formal contexts and constructing concept lattices using the OWLFCAView Protégé plugin.

Three scenarios are identified for formal context generation:

1. **Product Class-Attribute Relationships**: Product classes correspond to objects in the formal context, attributes correspond to properties, and the relation indicates class-property possession. For example, under Class 34-34 "Medical Equipment (Devices and Small Medical Equipment)," all attributes are stored under the "Property" category. The axiom "has_property some values from Property" establishes relationships between product classes and their attributes. [Figure 4: see original paper] illustrates a partial formal context for product class-attribute relationships, where rows represent classes, columns represent attributes, and "×" indicates attribute possession. [Figure 5: see original paper] shows the resulting concept lattice.
2. **Product Class-Part Relationships**: Product classes correspond to objects, parts correspond to attributes, with relations indicating component possession. Under Class 34-34, parts include "light source," "examination instrument," and "transmission medium." The axiom "has_parts some values from Parts" generates the formal context shown in [Figure 6: see original paper] and concept lattice in [Figure 7: see original paper].
3. **Product Class-Accessory Relationships**: Similar to parts but with different semantics. Using the axiom "has_accessories some values from Accessories," [Figure 8: see original paper] and [Figure 9: see original paper] present the formal context and concept lattice for accessories.

4.3 Ontology Prototype Generation and Formalization The OWLFCAView Protégé plugin converts concept lattices into ontology conceptual models, producing initial prototypes. After domain expert review and refinement, Protégé generates the final medical product ontology and corresponding OWL DL documents. The following XML snippet demonstrates the formalized representation:

```
<?xml version="1.0"?>
<rdf:RDF xmlns:xsp="http://www.owl-ontologies.com/2005/08/07/xsp.owl#"
         xmlns:swrlb="http://www.w3.org/2003/11/swrlb#"
         xmlns:swrl="http://www.w3.org/2003/11/swrl#"
         xmlns:protege="http://protege.stanford.edu/plugins/owl/protege#"
         xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
         xmlns="http://www.owl-ontologies.com/Ontology1430730323.owl#"
         xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
         xmlns:owl="http://www.w3.org/2002/07/owl#"
         xml:base="http://www.owl-ontologies.com/Ontology1430730323.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Tongue_depressor">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom>
```

```
          <owl:Class rdf:ID="Examination_or_measuring_instrument"/>
        </owl:someValuesFrom>
        <owl:onProperty>
          <owl:ObjectProperty rdf:ID="is_accessory_of"/>
        </owl:onProperty>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:someValuesFrom>
          <owl:Class rdf:ID="Diagnostic_instrument_set"/>
        </owl:someValuesFrom>
        <owl:onProperty rdf:resource="#is_accessory_of"/>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Class rdf:ID="Accessories"/>
    </rdfs:subClassOf>
  </owl:Class>
</rdf:RDF>
```

This snippet formalizes "Tongue Depressor" as a subclass of Accessories and defines its accessory relationships with "Examination or Measuring Instrument" and "Diagnostic Instrument Set."

5. Conclusion

This study explores product ontology construction by reusing product classifications through concept lattice theory, proposing a robust methodology that significantly differs from traditional approaches. The method provides a formal framework integrating concept lattices into the product ontology construction process, enabling formalized and visualized representation of product/service information to meet enterprise information management needs.

Key insights include:

1. **Domain and Scope Clarification**: Clearly defining ontology domain and scope before construction reduces redundancy. While eCl@ss provides comprehensive attributes, their cross-category intersections can cause redundancy if directly reused. Defining object properties to link classes with attribute dictionaries clarifies these relationships.
2. **Scientific Classification**: Systematically organizing categories according to scientific principles improves hierarchical rationality. Despite eCl@ss' s scalability limitations due to its four-level hierarchy, pre-construction classification reorganization enhances logical consistency.

3. **Property Dictionary Construction**: Building attribute dictionaries and defining object properties eliminates attribute redundancy. Three relationship types were identified: class-attribute, class-part, and class-accessory, each formalized through corresponding object properties.
4. **Formal Context and Concept Lattice Generation**: This approach enhances ontology reusability and sharability through formal semantics. Formal contexts, composed of binary relations between objects and attributes, provide formal information models. Concept lattices, as Hasse diagrams, visually present entity relationships, overcoming the lack of formal semantics in traditional natural language-based ontology construction.

Future research directions include: (1) validating the method across different domains to assess practical applicability; (2) developing product ontology mapping methods for heterogeneous ontology integration; and (3) extending the methodology to other standard product classifications such as UNSPSC and GPC.

References

[1] Wille R. Restructuring Lattice Theory: An Approach Based on Hierarchies of Concepts [A] // Ivan Rival. Ordered Sets [M]. Netherlands: Springer, 1982: 445-470.

[2] Xie Zhipeng, Liu Zongtian. Concept Lattice and Association Rule Discovery [J]. Journal of Computer Research and Development, 2000, 37(12): 1415-1421.

[3] Bi Qiang, Teng Guangqing. Analysis of the Progress and Hotspots in Applied Research of FCA and Concept Lattice Theory Abroad [J]. New Technology of Library and Information Service, 2010 (11): 17-23.

[4] Hu Keyun, Lu Yuchang. The Concept Lattice and Its Application Progress [J]. Journal of Tsinghua University: Science and Technology, 2000, 40(9): 77-81.

[5] Haav H M. A Semi-Automatic Method to Ontology Design by Using FCA [C] // Proceedings of the 2nd International Workshop on Concept Lattices and their Application, 2004: 189-207.

[6] Cimiano P, Hotho A, Stumme G, et al. Conceptual Knowledge Processing with Formal Concept Analysis and Ontologies [C] // Proceedings of the 17th International Conference on Industrial and Engineering Application of Artificial Intelligence and Expert System. 2004: 189-207.

[7] Liu Ping, Hu Yuehong. Development of Domain Ontology in Information Science Based on FCA and Association Rules [J]. New Technology of Library and Information Service, 2012(2): 34-40.

[8] Nanda J, Simpson T W, Kumara S R, et al. A Methodology for Product Family Ontology Development Using Formal Concept Analysis and Web Ontology Language [J]. Journal of Computing and Information Science in Engineering, 2006, 6(2): 103-113.

[9] Ouyang Chunping, Hu Changjun, Li Yang, et al. Approach of Ontology Learning from Relational Database Based on FCA [J]. Computer Science, 2011, 38(12): 167-171.

[10] Teng Guangqing, Bi Qiang. A Study on Domain Ontology Construction from Heterogeneous Resources Based on Concept Lattice [J]. New Technology of Library and Information Service, 2011(5): 7-12.

[11] Huang Qi, Zheng Jianguo. Ontology Query for Product Classification Based on Semantic Analysis [J]. Journal of the China Society for Scientific and Technical Information, 2015, 34(4): 398-413.

[12] Noy N F, McGuinness D L. Ontology Development 101: A Guide to Creating Your First Ontology [R]. Stanford: Stanford Knowledge Systems Laboratory Technical Report, 2001: 1-25.

[13] Uschold M, Gruninger M. Ontologies: Principles, Methods and Applications [J]. Knowledge Engineering Review, 1996, 11(2): 93-136.

[14] National Cancer Institute. OWLFCAView Manual [EB/OL]. [2015-05-02]. https://wiki.nci.nih.gov/display/Protege/OWLFCAView+Protege+Tab+Plugin+Manual.

[15] eCl@ss. eCl@ss Wiki [EB/OL]. [2015-06-08]. http://wiki.eclass.eu/wiki/Main_Page.

[16] Gu T, Wang X H, Pung H K, et al. An Ontology-Based Context Model in Intelligent Environments [C] // Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference. 2004: 270-275.

Author Contributions: Lu Jiaying, Yuan Qinjian, Huang Qi, and Qian Yunjie conceived the research idea, designed the methodology, and finalized the manuscript. Lu Jiaying conducted the experiments, collected and analyzed data, and drafted the paper.

Conflict of Interest Statement: The authors declare no competing financial interests.

Supporting Data: Available at http://www.infotech.ac.cn
[1] Lu Jiaying, Yuan Qinjian, Huang Qi, Qian Yunjie. eclass_medical.rdf-xml.owl (Domain ontology OWL document from case study).
[2] Lu Jiaying, Yuan Qinjian, Huang Qi, Qian Yunjie. eclass_class_adjustment.doc (Adjustment plan for 34 major categories).
[3] Lu Jiaying, Yuan Qinjian, Huang Qi, Qian Yunjie. eclass_34_ch-en.csv (Chinese-English mapping of 34 major categories).

Received: 2015-12-27
Revised: 2016-02-28