Research on the Construction of an Ontology-Based National History Knowledge Retrieval Platform

Authors: Wang Ying, Zhang Zhixiong, Sun Hui

Date: 2016-04-19T00:00:00+00:00

Abstract

[Purpose/Significance] This study aims to construct a national history knowledge retrieval platform to improve users' efficiency in accessing national history knowledge and promote national history publicity and education. [Method/Process] The paper proposes the conceptual design and overall framework for an ontology-based national history knowledge retrieval platform. Based on a constructed national history ontology knowledge base, it employs Neo4j database as RDF data storage and creates instance indexes, triple indexes, and term indexes using Solr. For diverse retrieval requirements, it designs and implements the retrieval engine's execution flow, retrieval statement construction methods, and query processing algorithms, while also designing visualization methods for national history knowledge presentation. [Results/Conclusion] The constructed national history knowledge retrieval platform provides retrieval and browsing services including entity retrieval, question answering, relational retrieval, temporal retrieval, and semantic resource browsing. The platform framework and key technical implementation schemes can serve as important references for deep retrieval services oriented toward domain knowledge.

Full Text

Construction of a Knowledge Retrieval Platform Based on Historical Ontology of the People's Republic of China

Wang Ying[1], Zhang Zhixiong[1], Sun Hui[2], Lei Feng[2]
[1](National Science Library, Chinese Academy of Sciences, Beijing 100190, China)
[2](Institute of Contemporary China Studies, Beijing 100009, China)

Abstract

[**Purpose/Significance**] This study constructs a knowledge retrieval platform for the history of the People's Republic of China (PRC) to enhance users' efficiency in accessing national history knowledge and to promote its publicity and education. [**Method/Process**] We propose a construction approach and overall framework for an ontology-based national history knowledge retrieval platform. Building upon a constructed national history ontology knowledge base, we employ the Neo4j graph database as RDF data storage and create three types of indexes based on Solr: instance index, triple index, and text item index. To address diverse retrieval requirements, we design and implement the retrieval engine's execution workflow, query construction methods, and query processing algorithms, complemented by visualization solutions for knowledge presentation. [**Result/Conclusion**] The national history knowledge retrieval platform has been successfully built, providing services including entity retrieval, question answering, relevance search, temporal retrieval, and semantic resource browsing. The platform's architecture and key technical implementations offer valuable references for developing domain-oriented deep retrieval services.

Keywords: ontology; entity search; question answering; relevance search; visualization
Classification Number: G250.7

1. Introduction

With the widespread adoption of the Internet, national history publicity and education websites have become crucial channels for promoting and teaching PRC history. Traditional portals such as the "History of the People's Republic of China Education Network" present China's founding journey, achievements, and experiences through navigational structures, web browsing, and multimedia content. Despite abundant historical information, learners face information overload. Even with full-text search capabilities, they cannot directly obtain the specific knowledge they need. Therefore, constructing a dedicated retrieval and browsing platform for national history knowledge is essential to enhance the intellectual depth, readability, and interactivity of the "History of the People's Republic of China Education Network," enabling users to learn about national history intuitively and achieve the goal of drawing lessons from history to guide present governance and educate future generations.

In recent years, foreign institutions have introduced innovative knowledge engine systems such as Kngine and WolframAlpha. Kngine delivers more meaningful knowledge search results by understanding semantic information of keywords or concepts, answering user questions, discovering relationships between concepts, and linking disparate data sources [1]. WolframAlpha provides direct answers to queries—for instance, when asked "How tall is Mount Everest," it not only returns the elevation but also displays the mountain's geographic location, nearby towns, and various charts [2]. To help users obtain information more quickly and simply, traditional search engine companies like Google, Baidu, and Sogou are transitioning from information search to knowledge search. Google introduced the Knowledge Graph feature to better understand user queries and directly present relevant information on the search results page, eliminating the need to visit source websites [3]. Baidu has developed entity search and relevance recommendation functions using box computing technology, such as searching for "movies similar to Inception." Sogou's Zhilifang search performs query semantic understanding and can infer that "Yao Ming's wife's height" is "190cm" while providing profile information about Ye Li and Yao Ming's relationship graph. The foundation of these innovative applications is the construction of large-scale knowledge bases containing entities and related facts. For example, Google extracts professional information from Freebase, Wikipedia, and other sources to build its Knowledge Graph, which already contained over 570 million entities by 2012 [3]. Sogou Zhilifang extracts information from semi-structured web pages, extracts entities and attributes from textual data, and integrates heterogeneous data with structured data to construct its ontology knowledge base [4].

Moreover, the Internet is evolving from a Document Web comprising only web pages and hyperlinks into a Data Web containing vast amounts of data describing various entities and their rich relationships [5]. Large volumes of semantic data such as RDF and OWL repository data have been published, and the scale of the Linked Open Data (LOD) cloud continues to grow, enabling the development of more intelligent retrieval applications. For example, the Sindice search engine crawls and collects RDF data from the semantic web, providing entity retrieval and query services for the linked data web [6]. The Semplore search engine employs a hybrid retrieval mechanism combining keyword and structured search, creating three types of inverted indexes (keyword, concept, and relationship) to provide retrieval and faceted navigation services for linked data [7]. FREyA is an ontology-oriented interactive natural language query interface that uses syntactic parsing and ontology-based queries to interpret user questions, leverages user feedback to resolve ambiguities, and constructs SPARQL queries to complete question answering [8]. Treo integrates entity search, spreading activation search, and Wikipedia-based semantic relatedness calculation to retrieve linked data, performing semantic matching between parsed user queries and dataset representations [9]. MEANS is a medical question-answering system combining natural language processing and semantic web technologies; it uses NLP techniques for deep analysis of medical questions and documents, creates RDF annotations for documents, converts user questions into SPARQL queries, and enables question answering over medical document collections [10].

Building upon these existing studies and systems, this paper proposes a construction approach for an ontology-based national history knowledge retrieval platform, introduces its overall framework, elaborates on key technical implementations, and finally demonstrates the platform's effectiveness.

2. Platform Design

2.1 Design Approach

The objective of the national history knowledge retrieval platform is to provide retrieval, browsing, and navigation functions based on a constructed national history ontology knowledge base, enabling users to conveniently read and explore collected and curated national history knowledge. To achieve this goal, we propose the following design principles:

1. Establish effective storage and flexible access mechanisms for the national history ontology knowledge base to enable semantic organization and further utilization of national history knowledge.
2. Leverage the structured semantics and inference rules of the ontology to achieve fine-grained revelation and potential knowledge mining of national history knowledge.
3. Provide entity-oriented retrieval and browsing functions for national history knowledge, supporting the discovery of associations between entities.
4. Implement question-answering capabilities that allow users to pose questions about national history and receive direct answers rather than relevant historical text materials.
5. Visualize national history knowledge associations through knowledge graphs to facilitate users' intuitive understanding of relevant information.
6. Enable jump browsing of knowledge entities through semantic linking of textual resources, allowing users to conduct extended reading.

2.2 Overall Framework

Based on the design approach, the overall framework of the national history knowledge retrieval platform is established as shown in Figure 1

, comprising three layers: data layer, function layer, and service layer. The data layer provides fundamental functions such as data storage and access. It stores the constructed national history ontology knowledge base and collected national history education text resources, and builds instance indexes, triple indexes, and text item indexes to support retrieval applications. It also provides data access interfaces for the storage and indexes, enabling the retrieval engine to access data. The function layer consists of the national history knowledge retrieval engine, which performs core functions including query analysis, retrieval scheduling, and result calculation. The retrieval engine parses user input, recommends relevant search terms, constructs appropriate retrieval expressions based on parsing results, and executes retrieval tasks. During retrieval, the engine utilizes the ontology's inference mechanism to mine implicit knowledge and expand retrieval scope, and performs statistical analysis and ranking on returned results. The service layer provides fundamental retrieval services including entity retrieval, question answering, relevance search, full-text search, and temporal retrieval. The retrieval engine executes searches and enhances users' knowledge acquisition efficiency through visual displays of national his-

1986 Bulletin of the Chinese Academy of Sciences Inaugural Issue

Pioneering a Golden Age for the Development of Science and Technology

—Inaugural Words

Lu Jiaxi
(Academician of the Chinese Academy of Sciences,
President of the Chinese Academy of Sciences)

Any publication is a tool for information exchange, in thincoroinnd exclongs into. This proper roron in lnid and written, physired is an affinate to trait forcark in brockinite form—and measurek to produce excarinology. Walo zyth, idergiht is as fit ao fleat alt , head iid as sheiviid at easly, therain inter nrearahdopows, bras in in to fo:th in is bitic way. Et is a mon, on projec in wild, propper technology, cocoan aciss, and tecid pro ocir ist beliest ties, science in indrorí in ITU an hisys, im to ikdy—th filnwngere in recently, it, isn't geat. Ir.. THd risk "Chinese sing to" liods is includes.

The propication in 120 pays, girote the selelaha, iaut:hirz coin at teward on furse way. He left with partlin in nend aruly. It is indut "bir fil" as irind in joune. Wae A to j' ze no te fies to explosium, safnod gath, & asalave erages their in shuricked morrings. "Bir fil" in kationalit &... IL A A IT firtir folnale of it: a, to help of fit as, romain kt ix fleah atroclits, fa fib T'Plinate th árstancad flecheb ik in to be gerxdo ln in... "Bir fil" is faise form to lis lit im, ic of bl uk hred at to fir 2 from iis to firish— n go to, in to honyt declead:finn spectages in science.

My nami is ut to in hong, haut tech is "bir fil" in zeru withæeids. It fas—to inverst dete them, beth to—fleez ic, or st to netus technology, id it, into zeards, will lutch to technology.

The compriates laerny at pane, hold thich cz hebysz cken at iit in is:żybi in europe and doniting by the proierernding diatarnhh. Here is "develting it in to technologyis, hi iy in, and directionalsath itc" people "A zir judituh in tmag. Rith m:xt or desp sk as in vt iti at lfe, nhek to in aznuaura.". Whnt did comp its to f ax yens , fit rt-se ty ic., Ill unot rose iċ iit toch tonalhy th sr śnsh ct ife is soegance as thez imin-śe as, solutions, its firtd or bailero zoiit co indane as, werd as insxtly in bwaxer, avend to azsipth, opoa, hekas kin in in fact fits.

Second blacks in its thme. In do is ide, its be cerp ao troct sfrocred tob — be challenged zdent in aegrantis al deth en logg edusatum. Thiit it could kid blay in—th th its, they dy "—clbdo ", in

Figure 1: Figure 1

tory entity knowledge graphs and semantic browsing of text resources, while revealing potential knowledge and deriving implicit knowledge to support further knowledge discovery.

3. Key Technology Implementation

3.1 Construction of National History Ontology Knowledge Base

The national history ontology knowledge base forms the foundation of the retrieval platform. Drawing upon experiences from other historical ontology constructions and considering the characteristics of the national history domain, we propose an approach and methodology for building the national history ontology and construct its conceptual model [11]. The ontology standardizes and semantically represents knowledge entities, concepts, and their relationships, including persons, institutions, meetings, and events. The national history ontology defines 19 classes (e.g., Event, Meeting, Person, Institution, Document, Concept/Term) along with 20 data properties and 76 object properties, and specifies some property constraints and inference rules based on common-sense knowledge. This establishes the classification of national history knowledge entities and types of entity relationships to guide the construction of the knowledge base.

The primary knowledge sources for the national history ontology knowledge base are foundational reference books for national history education, such as the *Encyclopedia of the History of the People's Republic of China* and the *Dictionary of the History of the Communist Party of China*. Since manually creating national history knowledge entities and relationships is time-consuming and labor-intensive, we employ a method combining automatic processing and manual curation to extract explicit national history knowledge from texts. First, domain experts select important national history terms as the basis for knowledge extraction, compile thematic vocabularies such as person and institution tables, and extract critical metadata including term names, types, and periods from reference books. Through manual organization and verification, these are transformed into foundational ontology instances and basic data properties. These instance data are then used to automatically annotate collected national history terms, extracting potential relationships or attribute data as auxiliary information for manual processing. Under the constraints of the national history ontology conceptual model, experts manually revise or construct ontology instances, edit instance properties and relationships, and review and manage the curated data to gradually build a national history ontology knowledge base containing important knowledge. Currently, the knowledge base comprises 15,602 instances, 5,147 property values, and 21,503 instance relationships.

3.2 Neo4j Storage

Given the characteristics of the national history ontology knowledge base, we selected the graph database Neo4j as the underlying data storage to support complex retrieval requirements such as structured and relevance searches. Neo4j is a high-performance NoSQL graph database that replaces traditional table designs with efficient graph data structures, storing data as graph nodes and relationships between nodes while providing graph traversal and lookup functions. Neo4j offers excellent scalability and flexibility, making it suitable for managing and querying complex relationships and supporting inference, which aligns with the triple storage and SPARQL query requirements of RDF graph data model-based ontology knowledge bases.

In the Neo4j database, each instance in the national history ontology knowledge base is stored as a node, instance data properties are stored as node properties and values, and instance relationships are stored as binary directed relationships between nodes. For example, when storing the instance "Mao Zedong," a new node is created in Neo4j with an automatically generated node ID of 327. Based on the instance's information in the knowledge base, the node property "label" is defined with the value "Mao Zedong" and the property "altLabel" with the value "Chairman Mao." Additionally, according to relationship types defined in the national history ontology such as "has held position" and "party affiliation," node relationships are defined as relationship[328]: has held position("Mao Zedong", "President of the People's Republic of China 1949") and relationship[326]: party affiliation("Mao Zedong", "Communist Party of China").

3.3 Index Design

We adopt Apache Solr indexing technology to build indexes for instances, property relationships in the national history ontology knowledge base, and national history education text resources, thereby improving query speed and system response time. Based on retrieval requirements, we construct three types of indexes:

1. **Instance Index**: The instance index enables rapid entity retrieval and faceted navigation for national history entities. Primary fields include: id, canonical name (label), alternative names (altLabel), initials, entity type (entityType), and related text item ID (textItemID). When users input search terms, the retrieval engine queries the instance index in real-time to provide term suggestions. After parsing user input, it performs exact and fuzzy matching using the label and altLabel fields to return matching or recommended instances, and enables faceted navigation based on fields such as entity type and initials.

2. **Triple Index**: The triple index supports fast retrieval of the national history ontology knowledge base by indexing its triples. Primary fields include: subject ID (sID), subject name (sLabel), subject type (sType), subject type name (sTypeValue), predicate ID (pID), predicate name (pLabel), predicate type (pType), predicate type name (pTypeValue), and object ID (oID), object name (oLabel), object type (oType), object type name (oTypeValue), and source text item ID (textItemID). The triple

index construction provides the foundation for structured relationship retrieval, enabling the retrieval engine to implement national history ontology knowledge base retrieval using Solr query expressions similar to SPARQL.

3. **Text Item Index**: The text item index targets collected national history education text entries to support full-text retrieval. Primary fields include: text item ID, title (itemTitle), content (itemText), type (itemType), and source database (itemSource).

3.4 Retrieval Engine Function Implementation

To meet diverse user retrieval needs, the national history knowledge retrieval platform provides three retrieval entry points: general retrieval, relevance retrieval, and temporal retrieval. General retrieval allows users to input search terms or statements to query relevant national history knowledge or pose questions to obtain answers. Relevance retrieval enables users to input two search terms to discover inter-entity relationships. Temporal retrieval is used to search for events, meetings, documents, etc., within a specified time range.

3.4.1 Retrieval Execution Process For the three retrieval entry points, the retrieval engine establishes corresponding processing methods based on input type, retrieval target scope, and output results. The specific execution process is illustrated in Figure 2 [FIGURE:2].

The general retrieval entry accepts arbitrary search terms to query target entities' related national history knowledge or accepts questions to obtain answers. The retrieval engine first processes general retrieval input using natural language processing techniques to extract entity terms and predefined variant terms from the national history ontology knowledge base. Based on matching results, it constructs instance retrieval expressions, full-text retrieval expressions, relationship retrieval expressions, or relevance retrieval expressions. If multiple entities share the same name or no matches are found, candidate entities and fuzzy-matched recommendations are presented to users for selection before constructing corresponding retrieval expressions.

The relevance retrieval entry requires users to input two entity names. Similarly, if the entity recognition identifies two distinct entities in the knowledge base, it directly constructs a relevance retrieval expression for execution. If only one entity is matched, users are prompted to convert to entity retrieval. If multiple or no entities are matched, candidate entities and fuzzy-matched recommendations are returned for user selection before constructing the retrieval expression.

Temporal retrieval requires users to input start and end years, select return types such as events, meetings, or documents, and the retrieval engine constructs corresponding retrieval expressions based on these selections.

After retrieval expression construction, the retrieval engine selects specific indexes or databases to execute retrieval tasks, returns results, and performs statistical analysis and ranking.

3.4.2 Retrieval Expression Construction The construction of the national history ontology knowledge base transforms national history knowledge into computable structured data. Using query methods similar to SPARQL enables fine-grained retrieval services. After query parsing, the retrieval engine constructs different retrieval expressions based on requirements:

1. **Instance Retrieval**: Instance retrieval queries the instance index using instance names, alternative names, types, and other fields to return exact or fuzzy-matched instance results, with sorting by name or relevance. For example, the retrieval expression for exactly matching the instance named "One Country, Two Systems" is:

   ```
   query=(label: "One Country, Two Systems") OR (altLabel: "One Country, Two Systems")
   ```

   The retrieval engine passes this expression to the instance index server for execution, then encapsulates and processes the results for front-end display.

2. **Relationship Retrieval**: Relationship retrieval primarily targets the triple index. Given an instance in the national history ontology knowledge base, it queries related properties and relationships by specifying the subject (s) or object (o) in the triple (s,p,o) and returns the triple result set. For example, when a user inputs "Mao Zedong" in the general retrieval entry, the retrieval engine first parses the query, obtains the matched entity's instance ID as 6787, and constructs the relationship retrieval expression as:

   ```
   query=(sID:6787) OR (oID:6787)                                    (2)
   ```

   If both an instance and a specific property are known, the system queries for instances or property values connected through the specified relationship by defining both s and p or p and o in the triple (s,p,o). For example, to query "participants" of "The 3rd Plenary Session of the 11th CPC Central Committee," the retrieval engine constructs:

   ```
   query= (sLabel:"The 3rd Plenary Session of the 11th CPC Central Committee") AND (pLabel
   ```

   According to domain characteristics, the national history ontology defines property constraints and inference rules. During retrieval expression construction, the retrieval engine reconstructs expressions using ontology reasoning mechanisms. In the above example, since the ontology defines inverse relationships between "participant" and "participate in," and a rule that "speakers and reporters at a meeting must have participated," the retrieval expression is reconstructed as:

   ```
   query=((sLabel:"The 3rd Plenary Session of the 11th CPC Central Committee") AND (pLabel
   ```

This discovers implicitly expressed national history knowledge and expands relationship retrieval scope.

3. **Relevance Retrieval**: Relevance retrieval leverages the graph structure formed by triple sets in the national history ontology knowledge base. Given two different instances, it queries paths between their corresponding graph nodes to obtain direct or indirect relationships. The system provides relevance degree options of "near," "moderate," and "far," allowing users to query paths with lengths no greater than 2, 3, or 4, respectively, returning triple sets along these paths. Relevance retrieval execution utilizes Neo4j's graph traversal mechanism through Cypher queries. For example, to retrieve "moderate" relevance between "Deng Xiaoping" and "The 3rd Plenary Session of the 11th CPC Central Committee," the retrieval engine first performs entity recognition to obtain entity IDs 5904 and 14563, then constructs the Cypher query:

```
start a=node(*), b=node(*) match p=a-[*0..3]-b where a.source_{id}=5904 and b.source_{i
```

The retrieval engine connects to the Neo4j database to read and parse the query results for system output.

4. **Full-Text Retrieval**: Full-text retrieval targets national history education text resources, performing searches on entry titles and content using the search terms. It employs Solr integrated with the mmseg4j Chinese word segmentation tool to complete full-text retrieval tasks and returns relevant entries sorted by relevance. For example, to retrieve entries related to "The 3rd Plenary Session of the 11th CPC Central Committee," the retrieval expression is:

```
query= (itemTitle:The 3rd Plenary Session of the 11th CPC Central Committee) OR (itemTe
```

5. **Temporal Retrieval**: Temporal retrieval primarily relates to time-based queries, such as searching for events, meetings, or documents published within a time interval. It queries triples where the subject type is Event, Meeting, or Document (including files, works, newspapers, reports, speeches, etc.) and the object type is Time. Since historical records often contain relatively vague time values, the original values are preserved during instance construction. To facilitate computational retrieval, the platform defines start time and end time properties for the Time class with granularity down to the month. During index construction, conversion rules map vague temporal expressions such as "first half of year," "spring," and "year end" to specific year-month values. For example, the retrieval expression for events occurring between 1949 and 1950 is:

```
query=(sTypeValue:"Event")  AND  (oTypeValue :"Time")                (7)
```

3.4.3 Query Processing Algorithm The general retrieval entry accepts both search terms and question statements, making its query processing more complex than relevance or temporal retrieval. This paper proposes a query processing algorithm. The retrieval engine first parses the user input string. Since national history knowledge entity names are typically Chinese with a minimum length of two characters, if the input contains fewer than two characters, instance retrieval is executed to recommend candidate entities to users. Otherwise, natural language processing techniques are applied, using instance names and aliases from the knowledge base as a dictionary to annotate entities in the input. If more than two entities are matched, the query is converted to full-text retrieval of text entries related to the input. If exactly two entities are matched, the query is converted to relevance retrieval between the two entities. If only one entity is matched, the query is converted to relationship retrieval for that entity, returning all related triples. If no entities are matched but variant terms are identified, the system maps variant terms to specific ontology properties or classes using a predefined mapping table. For example, "when was it held," "when," and "held time" are mapped to the ontology property "held time," while "region," "place," and "where" correspond to the ontology class "country and region." Further analysis based on property and class lists determines the retrieval type. When one entity is matched and the property list is non-empty, retrieval is converted to relationship retrieval for the entity's specified properties. Otherwise, based on the class list, relationship retrieval returns triples between the matched entity and entities belonging to the specified class. If both property and class lists are empty, entity triple relationship retrieval is performed. When no entities but variant terms are matched, if the variant term corresponds to a class list, retrieval is converted to instance retrieval for entities belonging to that class; otherwise, it is converted to full-text retrieval of the input string. This approach leverages entity and variant term vocabularies to achieve natural language processing of queries, forming diverse retrieval expressions for specific scenarios to retrieve national history knowledge from both the knowledge base and text resources. The algorithm is shown in Table 1 .

3.5 Visualization Implementation

To intuitively present relationships among national history knowledge entities, we selected Cytoscape Web for knowledge graph visualization. Cytoscape Web is an open-source graph visualization library whose data model supports nodes and directed edges, allowing definition of node and edge names and types to meet application requirements for displaying knowledge base instances and relationship names. Cytoscape Web's network display components are implemented through Flex/ActionScript, offering diverse visualization styles and providing JavaScript APIs for customizing network views and interactions, supporting zooming, dragging, click events on nodes and edges, and type filtering.

In the project, Cytoscape Web's JavaScript files are imported, and backend programs encapsulate visualization data including node and edge colors, names, types, and display styles, generating JSON format data transmitted to the front-end for rendering. Custom jQuery methods parse this data, configure graph styles, implement Ajax operations, and invoke Cytoscape Web's JavaScript methods to draw graphs and handle events.

In the visualization window, nodes represent entities, directed edges between nodes reveal entity relationships, and different colors represent entity classes, intuitively presenting the knowledge graph. Rich user interaction features are provided, such as moving, zooming, and fitting the graph to window size through a floating control panel. Clicking and holding on blank space transforms the cursor into a hand icon for dragging the entire graph, and individual nodes can also be dragged. Left-clicking a node enables browsing its knowledge graph, right-clicking displays detailed node information, left-clicking an edge shows the relationship name on the graph, and right-clicking an edge reveals the knowledge source (source text entry) for that relationship. Clicking the ontology class color legend on the right side of the window filters entity types within the view. The system supports visual browsing, allowing users to continuously acquire relevant national history knowledge by clicking nodes in the knowledge graph. As shown in Figure 3 [FIGURE:3], when browsing the knowledge graph for "The 3rd Plenary Session of the 11th CPC Central Committee," participants are displayed. Clicking a participant such as "Hua Guofeng" reveals related knowledge, showing that the event "China-US diplomatic relations" connects to "Hua Guofeng" through the "related person" relationship. Clicking the "China-US diplomatic relations" node further reveals related persons, events, documents, and sub-events. The system preserves browsing paths and history, allowing backward navigation. This achieves roaming-based browsing through knowledge graphs, facilitating direct and convenient access to national history knowledge.

4. Platform Implementation and Effects

Based on the above design and key technology implementations, we have completed construction of the national history knowledge retrieval platform. The platform is a B/S-mode web application system developed in Java using the SpringMVC and Hibernate frameworks. It employs Neo4j 2.1.2 Win64 (free version) as the database, Solr 4.7.2 for indexing, runs on Windows Server 2008, uses Tomcat 6.0.4 as the web server, and JDK 1.7. The platform provides entity retrieval, question answering, relevance search, temporal retrieval, and semantic resource browsing services.

4.1 Entity Retrieval

Unlike traditional retrieval that presents matching text resources, the national history knowledge retrieval platform visualizes knowledge about entities that match retrieval targets from the national history ontology knowledge base. Users can intuitively understand relevant national history knowledge without reading through textual information, while extended reading is enabled through knowledge graph interactions, making knowledge acquisition more efficient. For example, searching for the event "Land Reform Movement" returns the knowledge graph shown in Figure 4 [FIGURE:4], clearly displaying related meetings, events, documents, concepts/terms, persons, and institutions. The platform also returns relevant materials about "Land Reform Movement," allowing users to continue reading related national history education text resources.

4.2 Question Answering

To meet users' needs for natural language questioning, we designed and implemented a national history knowledge question-answering function. Using natural language processing techniques to analyze user questions, the system constructs structured retrieval expressions against the national history ontology knowledge base and returns knowledge graphs as results. For example, when a user asks "who proposed 'Imperialism and all reactionaries are paper tigers'," the system returns the knowledge graph shown in Figure 5 [FIGURE:5], automatically identifying that the "concept proposer" of "Imperialism and all reactionaries are paper tigers" is "Mao Zedong."

4.3 Relevance Retrieval

Relevance retrieval leverages the graph structure of the national history ontology knowledge base to discover inter-entity relationships, obtain subgraph structures, and mine potential knowledge. For example, searching for "near" relevance between "Chen Yun" and "Two Whatevers" yields the knowledge graph shown in Figure 6 [FIGURE:6]. The graph reveals that "Chen Yun" is associated with the meeting "Central Committee Work Conference (March 1977)" as a "speaker or reporter," that this meeting's "related concept or term" is "Two Whatevers," and that both the event "Deng Xiaoping's third comeback" and the event "Criticism and resistance to the 'Two Whatevers' from inside and outside the Party" are related to person "Chen Yun" and concept/term "Two Whatevers." Thus, the subgraph demonstrates the intrinsic connection between "Chen Yun" and "Two Whatevers."

4.4 Temporal Retrieval

Temporal retrieval allows users to select a time range to find events, meetings, documents, and other information. Searching for events and meetings between 1949 and 1950 returns the results shown in Figure 7 [FIGURE:7], including 67 events and 37 meetings arranged chronologically. Clicking an entity name enables further browsing of its knowledge graph.

4.5 Semantic Resource Browsing

The platform provides semantic resource browsing functionality on text entry detail pages. A text annotation program identifies important entities in the text, using different colors to distinguish types. Clicking an annotated entity name automatically jumps to that entity's knowledge graph page, facilitating further exploration of related national history knowledge. As shown in Figure 8 [FIGURE:8], the entry for "Land Reform Movement" annotates national history entities and concepts including persons, meetings, events, institutions, countries/regions, special groups, concepts/terms, and documents.

5. Conclusion

This paper constructs a national history knowledge retrieval platform based on national history ontology, exploring ontology knowledge base storage, indexing, visualization, and ontology-based knowledge retrieval technologies. Our research utilizes ontology to represent and organize national history knowledge at a finer granularity, implementing structured, fine-grained retrieval services including entity retrieval, question answering, relevance search, and temporal retrieval. The platform supports deep mining and exploration of national history knowledge, extends the depth of information retrieval, and improves knowledge acquisition efficiency. Furthermore, knowledge graph visualization and semantic resource browsing enrich result presentation formats, enhance user experience, and improve the interactivity and novelty of the "History of the People's Republic of China Education Network." Future work will continue to expand instances in the national history ontology knowledge base to enrich its content, while improving platform performance for large-scale data processing to further enhance service effectiveness.

References

[1] Kngine[EB/OL]. [2015-03-10]. http://www.baike.com/wiki/Kngine.
[2] WolframAlpha[EB/OL]. [2015-05-10] http://baike.baidu.com/link?url=7hCseBbIipm9Xo0eMo7jVuCrgcUM
yW7lUr9YkjUw7_{{{WmG3}}*{{i9yeyiRAoRVCvNCPbD6JxzNfyJ4ET6}}}.*
[3] Singhal, Amit. Introducing the Knowledge Graph: Things, Not Strings[EB/OL]. [2013-04-10]. http://googleblog.blogspot.co.uk/2012/05/introducing-knowledge-graph-things-not.html.
[4] Zhang Kuo. From Information Search to Knowledge Search—Technical Architecture[EB/OL].[2013-03-26]. http://weibo.com/1870490225/zbzDwq5TF#{rnd1435219297630}.
[5] Wang Haofen. Large-Scale Knowledge Graph Technology[EB/OL]. [2014-10-12]. http://blog.sciencenet.cn/home.php/fgcfmt/home.php?mod=space&uid=1225851&do=blog&id=801901.
[6] Tummarello G, Delbru R, Oren E. Sindice.com: Weaving the Open Linked Data[C]. Proceedings of 6th International Semantic Web Conference, Springer. LNCS 4825, 2007.
[7] Wang H. et al. Semplore: A Scalable IR Approach to Search the Web of Data [J]. Web Semantics: Science, Services and Agents on the World Wide Web, 2009, 7(3): 177-188.
[8] Damljanovic D, Agatonovic M, Cunningham H. FREyA: An Interactive Way of Querying Linked Data Using Natural Language[C] Proceedings of ESWC 2011 Workshops, LNCS 7117, 2012: 125-138.
[9] Freitas A. et al. Querying Linked Data Using Semantic Relatedness: A Vocabulary Independent Approach[C]. Proc. 16th Int'l Conf. Applications of Natural Language to Information Systems (NLDB 11), Springer, 2011, pp. 40-51.

[10] Abacha A, Zweigenbaum P. MEANS: A medical question-answering system combining NLP techniques and semantic Web technologies[J]. Information Processing and Management, 2015, 51:570-594.
[11] Sun Hui, Lei Feng. Preliminary Study on the Construction of Historical Ontology of the People's Republic of China[J]. Modern Intelligence, 2014, 34(2):32-42.

Author Contributions:
Wang Ying: Paper writing, research proposal, detailed system design and implementation.
Zhang Zhixiong: Research direction, paper revision.
Sun Hui, Lei Feng: Functional design, data processing.