# Analysis and Visualization of Narrative in Shanhaijing Using Linked Data

**Qian Wang**

A thesis submitted for the degree of
Bachelor of Advanced Computing (Honours) at
The Department of Computer Science
Australian National University

July 2019

Except where otherwise indicated, this thesis is my own original work.

Qian Wang
30 July 2019

To my grandparents, who are not in the world anymore.

# Acknowledgments

I would like to thank my parents for their support and encouragement. I would like to thank my supervisors, Dr Ben Swift and Dr Terhi Nurmikko-Fuller, for their guidance, patience and understanding throughout this project. Finally, I would like to thank my Honours cohort, for their company and help.

# Abstract

The Linked Data (LD) literature contains several examples of analysing Western folktales using LD methods. However, similar linked datasets and analysis for ancient Chinese mythology is (seemingly) non-existent. This thesis seeks to bridge that gap by creating, analysing and publishing LD for the classic Chinese mythology *Shanhaijing*. We first capture the unstructured information contained in the book as structured data, based on a close reading of the text in both English and classical Chinese. Then we develop a schema-level representation of the underlying ontological model which captures those data types and represents the relationships between them, which we use to populate the dataset using RDF, including instance-level information. Finally, we implement an LD-based interactive data explorer software tool for visualising, querying (through a SPARQL interface) and analysing the data to explore the relations and elements from the book. We have published the resulting dataset, ontology and source code to GitHub so that other scholars can benefit and incorporate this data into the wider LD Web.

# Contents

# List of Figures

# List of Tables

Chapter 1

# Introduction

## 1.1 Background

The LD paradigm is a mechanism for employing the Web as a means for data and knowledge integration wherein both documents and data are linked [Berners-Lee et al., 2001]. It offers an efficient way to process and publish structured data. It operates on RDF triples [RDF Working Group, 2014], established upon existing Web infrastructures, such as HTTP URIs [Berners-Lee, 2002], the RDF data model [Klyne et al., 2004], triplestores, and the SPARQL query language [Prud'hommeaux and Seaborne]. Data can be interlinked from different sources, increasing its value and usefulness, and making it readable for both human and machines. Existing datasets published as LD have covered many areas, ranging from entertainment to medical treatment.

In the domain of literature, several publicly available datasets have been published as LD. For example, the BookSampo project [Mäkelä and Hyvönen, 2011], which provides information on fiction literature published in Finland going back to the 15th century, alongside rich descriptions of both their content and context [Mäkelä et al., 2013]. The Perseids Project [Almas et al., 2017] provides a platform for creating, publishing, and sharing research data, in the form of textual transcriptions, annotations and analysis [Almas, 2017].

Much effort has been spent in analyzing Western literary traditions, especially mythologies, using LD methods. Some essential work in this scope includes the Brothers Grimm project [Franzini et al., 2015], Vladímir Propp's Morphology of the Folktale [Peinado et al., 2004], ancient Sumerian mythologies [Nurmikko-Fuller, 2014], and the Greek mythology [Syamili and Rekha, 2018], all utilizing LD techniques to examine Western folktales.

## 1.2 Motivations

Despite the work on the narrative of separate regions of some prominent Western myths (like the European, Russian, Sumerian, and Greek folktales), projects focused on Chinese literature within this interdisciplinary field are rare. Most of these projects concentrate on a single composition, *Dream of the Red Chamber* (红楼梦)

[Xueqin, 2011], which is one of the most well-known Chinese classic novels, written by Cao Xueqin(曹雪芹) during the 18th century. Those projects that have combined computational methodologies with Chinese literature have been more interested in adopting natural language processing (NLP) techniques [Liu et al., 2015], as well as taking 'Delta' [Burrows, 2002] as a way to analyse the book [Du, 2017], rather than employing LD methods.

An extensive literature review of research in connecting computing with folktales found only very few examples of analyses of ancient Chinese mythology. Work in this domain using LD methods to analyse the narrative is, in turn, seemingly non-existent.

Ancient Chinese mythology shares a similar status for Chinese and even for other Asian cultures as Western myth does for the Western world. For example, a classic Chinese compilation of mythic geography and myth, *Shanhaijing* (*the Classic of Mountains and Seas*, 山海经), which is an ancient encyclopedic cosmography that described a myriad of strange or imaginary creatures [Qiu, 2008], occupies a significant position and is representative of Eastern mythologies. Over thousands of years, numerous novels, literary fictions, dramas and plays have been derived from this book, and also a significant number of mythologies from other Asian countries were influenced by this text, such as the *Kaiki choju zukan* [尾崎勤, 2004] and *Hyakki Yagyō* [Takako, 1999], both Japanese folklore with an origin in *Shanhaijing*. Under this circumstance, a similar richness of analyses utilising computing methods needs to occur for using Chinese mythology.

Existing LD methods have been developed almost exclusively in the context of Western culture, and predominantly for highly structured data. When facing ancient Chinese mythologies, there still several unsolved challenges:

- **Nonexistent Structured Dataset** Currently, unlike the Brothers Grimm folk tales [Franzini et al., 2015], no equivalent or comparable work has been undertaken to identify the tropes as the motifs of Chinese tales. Very limited motifs of the Chinese mythic books have been labelled or recorded in the Aarne-Thompson Motif-index of folk-literature, which is a valuable classification tool of narrative elements in folktales, ballads, myths, fables, mediaeval romances, exempla, fabliaux, jest-books, and local legends [Thompson, 1989]. Few available datasets were containing the information of Chinese folk tales in a digital or structured format currently exists, which means that before LD methods can be applied to the narrative of Chinese myths, a structured dataset based on the in-depth knowledge of the Chinese mythology must be constructed.

- **Unavailable Ontologies** Ontologies play an important role in Semantic Web (SW) [Berners-Lee et al., 2001] applications [Ibrahim et al., 2013]. Ontologies define common vocabularies for researchers who need to share information in a domain [Alhawiti and Abdelhamid, 2016]. They include machine-interpretable definitions of basic concepts in the domain and relations among these concepts [Lau and Li, 2006].

  At present, some ontologies exist in the domains of Western folktales. For example, the ontology created for the heroes of ancient Greek Mythology [C and RV, 2017], the one generated from Propp's morphology of the folktale [Peinado et al., 2004], and the most famous one, Aarne-Thompson's Motif-Index. However, the narratives of Eastern and Western mythologies are not entirely similar. For example, in the Greek mythology, most gods and heroes are human-like, both physically and emotionally. However, in some famous ancient Chinese mythologies (such as *Shanhaijing*), numerous gods and monsters are described as a combination of different animals. For instance, a monster might look like a pheasant, but has a fish's body, five bird's heads and three serpent's tails. Based on the differences in the narratives, most ontologies created for Western folktales are not entirely suitable for the representation of ancient Chinese mythic classics and could not demonstrate the characters of gods and monsters adequately, although they do contain complementary aspects and part of them can be connected to an ontology produced for Chinese myths. For example, in the Aarne-Thompson's Motif-Index, a famous monster in Asian folk tales, Nine-tailed Fox (九尾狐) [Nozaki, 1961][Lee, 2011][Chen, 1995], is recorded at B15.7.7.1.. Other motifs consist of the four-eyed tiger (at B15.4.1.2.) and serpent with jewel in mouth (at B103.4.2.), etc. Such kind of strange creatures could be interlinked with the monsters of Chinese mythologies. However, because these ontologies neither contain the narrative of Chinese myths nor are created for Chinese folktales, those included examples are insufficient.

## 1.3 Contributions

Few ontologies in existence focus on Eastern folktales. To be more specific, almost no available ontology has been developed specifically for ancient Chinese mythology. This thesis seeks to bridge that gap with the ancient Chinese mythic book *Shanhaijing* (both the traditional Chinese version [Yuan, 1980] and the English translation [Birrell, 1999]), introduces ancient Chinese mythology to the broader public as well as a broader academic community, and contributes to the diversified culture of literary analysis.
In this thesis, we complete the analysis of the "monster" category of character types that appear in *Shanhaijing* and develop an interactive data explorer to demonstrate those data results. Our main contributions are listed below:

1. We rely on traditional humanities methodologies such as the broad reading to analyse and extract knowledge from selected resources[1], which we completed in both Chinese and English. We created a dataset categorising both the character types and the types of relationships between them. This process is described in Chapter 4.2.

2. With the use of some existing vocabularies, we produce an innovate ontological structure to represent those data categories. After satisfied with the theoretical

[1] those selected resources for this thesis are listed in Section 1.4

accuracy of the model, we implement it to produce a second original dataset, this one is in RDF, using existing software Protege [Musen, 2015]. With the help of LD technologies, we create a new ontology and vocabulary, which is, in turn, linked to other relevant ontologies. The detailed descriptions are involved in Chapter 4.4.

3. We use the generated ontological structure to map the instance-level data in our dataset through Protege, and interlink the mapped data to other similar available data both manually and automatically through SILK [Volz et al., 2009]. In which way the LD for the monster category of *Shanhaijing* is created. This work is demonstrated in Chapter 4.5.

4. We develop an ontology-based interactive LD explorer tool based on dotNet Framework to visualise the generated relations and elements, allowing end users to construct and run useful queries based on the contents of *Shanhaijing*, thus acquiring various detailed linked information about the book. This explorer is designed to be accessible for both users who are unfamiliar with or specialised in the underlying ontologies produced for the book and the SPARQL query language, providing live visualisations to explore the book. This tool is discussed in Chapter 5, and some use cases for the LD of this thesis are demonstrated in Chapter 6.

5. Our work has been published on an Open Access platform, GitHub[2], which involves the resulting dataset, the developed ontological structure, the SPARQL queries used to retrieve data from the system and the interactive data explorer. This allows future generations of scholars to benefit not only from our analysis and findings but also to be able to verify them by repeating the workflow on the same dataset. Future researchers will also be in a position to benefit from, reuse, and redevelop our processes, tools, and data for their research.

## 1.4 A Note on Language and Translation

We identified diverse sources containing the knowledge of the monster category of *Shanhaijing*, which cover books, blogs, historical studies, journals, comments, and encyclopedias etc., and are written in classical Chinese, vernacular Chinese and English language versions.
Having considered and compared these resources carefully, we select some of them as the research sources for this thesis, which are listed below:

- *Shanhai Jing with Collation and Annotation* (山海经校注) [Yuan, 1980] by Yuan Ke in classical Chinese;

- *Shanhai Jing Zhu* (山海经注) [Guo, 2017] by Guo Pu in classical Chinese;

[2]We published our work at the repository https://github.com/aaasteria/chinesemonster

- *Shanhai Jing with Annotation* (山海经笺疏) [Hao, 2016] by Hao Yixing in classical Chinese;
- *Shanhai Jing in vernacular Chinese* (白话山海经) [Zhang and Qin, 1997] by Zhang Yanyun in vernacular Chinese;
- *the Classic of Mountains and Seas* [Birrell, 1999] by Anne Birrell in English version;

These resources cover the three language versions and are representative publications in this domain. Because *Shanhaijing* was created two thousand years ago and was translated and published in several versions, there is some discrepancy among the chosen materials. For example, in some versions, `Likewingbird (比翼鸟)` lives to the east of `Ruincoverbird (灭蒙鸟)`, while others believe that it is at the east of `Mount South (南山)`. Regarding this issue, we consider all the resources, combine them with our understanding, and then determine the suitable terms for this thesis by ourselves.
Since this thesis is written in English, the representations of instances and entities in our ontology and LD are mainly based on the translation by Birrell [1999]. However, terms in this book are translated literally, in which way, specific meanings of the instance names are lost, and recognising terms by their names becomes challenging. To avoid this situation, not only do we adopt the English interpretations in [Birrell, 1999] to describe our LD instances, but we also add related Chinese presentations as the label of our data and switch the different language versions regarding our requirements.
In this thesis, when an example with a specific Chinese meaning appears at the first time, no matter it is an instance from *Shanhaijing* or a literary text or paper we referenced, we all represent it in English translations, Chinese words and *pinyin*[3]. This expression is determined for disambiguation as well as improving recognition. After that, concerning different situations, we will choose either English translations or *pinyin* as the presentation of the specific example. For instance, at the first time, we introduced *Shanhaijing* (*the Classic of Mountains and Seas*, 山海经) to express the domain of this thesis, but later the word *Shanhaijing* is adopted for the same object. The reason for selecting *Shanhaijing* as the representation is that comparing with the English translations, the word *Shanhaijing* is shorter, more novel and more recognisable.

## 1.5 Roadmap

The remainder of this thesis is organised as follows: Chapter 2 introduces the book, *Shanhaijing*, the focus of this thesis. Chapter 3 gives a detailed description of the basic concepts and definitions that are used throughout this thesis. Chapter 4 presents the process of generating the structured dataset, the ontological model, the visualisation of the produced ontology and LD. Chapter 5 describes the interactive data explorer

[3] *pinyin* is the 'spell-sound' Chinese

we created for the visualisation of elements and relations of the monster category of *Shanhaijing*. Chapter 6 discusses some specific use cases that we adopted for querying our LD via the interactive explorer. Chapter 7 evaluates the performance of our ontology and the data explorer, concludes this thesis and points out potential work based on this research in the future.

## 1.6 Conclusion

In this chapter, we introduced the background and motivations of this thesis, with a brief discussion of related work. The obstacles we faced are pointed out in Section 1.2, and potential solutions are provided as well. We also mentioned our contributions, involving a structured dataset, a domain-specific ontology and LD of *Shanhaijng*, as well as an interactive explorer tool. In Section 1.4, we talked about the translations in this thesis. Finally, Section 1.5 presents the outlines of this thesis.

Chapter 2

# Overview of Shan Hai Jing

In this chapter, we offer a brief overview of the Chinese mythological compilation *Shanhaijing*. Section 2.1 introduces the background of the book. Section 2.2 provides a summary of the contents of this compilation. Section 2.3 presents the status and influence of the book in China and other Asian countries, and we conclude this chapter in Section 2.4.

## 2.1 Background

*Shanhaijing* (see Fig. 2.1) is the first compilation of myth and legend in ancient China. Most scholars have agreed that it is an interpretation of the mythological graph, *Shanhaitu*. *Shanhaijing* experienced a long time before its completion, which can be traced back to the pre-Qin period of China (4th century BC) and went through the early Han Dynasty (1st century AD). It is an encyclopedia of ancient Chinese culture which preserves a large number of primitive myths. The exact time that the book was written and the specific authors of the book are not known. Contemporary scholars have reached a consensus that this book was not written by one person or at one time, but is a collective result accumulated by different authors at different times and collaborated at the Western Han Dynasty (about four hundred years in total). Existing versions of *Shanhaijing* are based on the work edited and revised by Liu Xiang and Liu Xin at the first year of Emperor Ai of Han (6 AD). The first annotation of the book appeared in the Jin Dynasty (2nd century AD) by Guo Pu. Guo's annotation is precise and concise, which is considered as the first milestone in the annotation history of *Shanhaijing*.

## 2.2 Contents

The present version of this book contains 18 chapters, which are divided into three categories: *Shanjing (Classic of the Mountains,* 山经*)*, *Haijing (Classic of the Seas,* 海经*)*, and *Dahuangjing (Classic of the Great Wilderness,* 大荒经*)*, approximately 31,000 words in total.

The systematic presentation of these myths differs in nature from other textual col-


Figure 2.1: One Page of *Shanhaijing* in Classical Chinese Version [Wu, 1983]

lections of ancient and early myths identified with other parts of the world, which are structured by some form of narrative continuity [Michael, 2001]. Compared with multiple works narrated across time, the content of *Shanhaijing* is structured with mythic geography, where elements are located and recorded regarding the place and direction. Most of the records start in the south, then move westward, then north, and finally to the middle of Jiuzhou (九州)[1]. Jiuzhou is surrounded by the East Sea, the West Sea, the South Sea and the North Sea. Unlike the modern convention, the order of the direction recording in this book is from south to east, which is related to the concept of the emperors facing south.

As an encyclopedia of ancient Chinese times, the content covers broad areas, such as ancient mythology, geography, property, witchcraft, religion, ancient history, medicine, folk customs, ethnic and other aspects [Hu, 2003]. *Shanhaijing* records ancient Chinese mythologies, many deities in this book have a profound influence on the development of Asian culture, such as Jingwei (精卫), Nüwa (女娲), Hanba (旱魃) and Xiwangmu (西王母).

It also records numerous mythological and mysterious monsters with fanciful or imaginary descriptions, including birds, beasts, dragons, and snakes, which are portrayed as possessing mythical powers or are possibly related to ancestor (totem) worship [Li and Chan, 2012]. For example, there is a monster named Stag-silkworm (鹿蜀, Lushu)[2], it lives in Mount Cherrysunny (纽阳之山). This monster looks like a horse with a white head, a scarlet tail and tiger's markings. The noise of this monster is similar to human crooning. It is said that whoever wears its fur will have

[1] Jiuzhou is the representative of ancient China

[2] This monster appears in the Chapter: *Classic of the Mountains: Southern*

more descendants. In the East Sea, there is a mountain called Mount Flowwave (流波山), a monster Awestruck (夔, Kui)[3] lives there, which looks like a black ox with one foot but no horn. When it comes out or goes back into the water, there will be wind and rain. The glare of the monster is similar to the sunlight, and the sound of which hears like thunder. Yellow Emperor (黄帝) killed it, using its skin to make a drum, with God Thunder (雷神)'s bone as the mallet. The drumbeat spread out five hundred miles, which frightened the world.

These descriptions of creatures to some extent help readers to understand the ancient ecological environment and the understanding and imagination of the ancients of unknown things. These strange and exotic animals also provide endless reverie for many art lovers and artists, and attract them to this classic world.

## 2.3 Influence and Status

### 2.3.1 For Chinese Literature

*Shanhaijing* represents the earliest systematic repository of the myths and deities from ancient and early China, and this work has served as the primary inspiration for all later Chinese mythology [Michael, 2001]. Also, many works of different literary genres, such as poems, lyrics, novels, and drama, have adapted tales from *Shanhaijing* [Li and Chan, 2012].

For instance, in *Zhongshanjing (the Classic of the Mountains: Central, 中山经)* chapter, Yan Emperor's youngest daughter died in the Mount Guyao (姑瑶之山), whose soul transforms into a patch of grass on the mountain, namely Yao Grass (瑶草). In *Zhuangzi* (庄子) [Zhuangzi and Palmer, 1996], The Yao Grass has evolved into the fable of the Goddess living on the Mount Gushe (姑射山). Similarly, the source of the Wushan Goddess Zhaoyun (巫山神女) in *Gaotang Fu* (高唐赋) [Song Yu, 1977] origins from *Shanhaijing* as well. Besides, she is also the 23rd daughter Yao Ji (瑶姬) of the Queen Mother (西王母) in Du Guangting's immortal book [Du, 1988]. Lin Daiyu, one of the heroines of the famous Chinese novel *Dream of the Red Chamber*, transformed from a crimson pearl mythic herb, inspires from *Shanhaijing* likewise.

Other literature, such as the Yuan drama *The Injustice to Dou'er* (窦娥冤) [Guan, 1980], the Ming novel *Romance of the Investiture of the Gods*(封神演义) [Xu, 2009], and the Qing composition *Strange Tales of Liaozhai* (聊斋志异) [Pu, 1988], all have links to Shanhaijing [Li and Chan, 2012].

### 2.3.2 In Other East Asian Folklore

Over thousands of years, this mythic text has influenced not only numerous novels, literary fictions, dramas and plays but also a significant number of mythologies of other Asian countries. Famous instances include the traditional Japanese folklores *Kaiki choju zukan* and *Hyakki Yagyō*.

*Kaiki choju zukan*, written during the Edo Period (17th century AD), depicts pictures

[3] Awestruck is from the Chapter: Classic of the Great Wildness: Eastern

of monsters based on *Shanhaijing*, combing with the Japanese culture system integration and the influence of multiple factors such as history, culture, society, and blending into some folk myths and legends.

*Hyakki Yagyō*, also called Night Parade of One Hundred Demons, is a concept in Japanese folklore since Henan Period (8th century AD), where quite a lot of the demons evolved from Chinese myths or literary monsters, and many of them refer to the monsters in *Shanhaijing*.

One prominent example is Hyakki Inugamike, which is a derivation from the monster of Quanrong Country (犬戎国) in the *Classic of Regions Within the Seas: Northern* (海内北经) chapter. Both of the two monsters are human-like dogs wearing clothes. They sit haughtily, with a girl kneeling down and serving them. The Merpeople are based on the monster of Diren Country (氐人国)[4] (海内南经) in *Shanhaijing*. Monsters in Diren Country have a human face but a body of fish, and they only have flippers without a foot. They are similar to the mermaids in Western folktales without good-looking appearances. When introduced into Japan, this became the original appearance of the Merpeople.

Others such as Kamaitachi has origins in the monster Qiongqi (穷奇) in the *Classic of Mountains: West* (Xishanjing, 西山经) chapter; Kwaakago shares similarities with the Malignforce-buzzard (蛊雕)[5], they disguise as a baby crying to attract people, thus killing and devouring human.

To this day, the East Asian monster culture has become more and more popular. In Japan, the monster culture has been one of the leading aesthetic symbols. As an essential branch of anthropology, monster culture has gradually grown as a topic of interest for numerous people. Also, the origin of it is *Shanhaijing*.

## 2.4 Conclusion

In this chapter, we first talked about the background and a brief history of *Shanhaijing*. We described the contents of this book, ranging from the narrative structure to the areas it covers. In Section 2.3, the influence and status of *Shanhaijing* are discussed, both for Chinese literature and other East Asian folktales.

[4]This monster comes from *Classic of Regions Within the Seas: Southern*
[5]Appears in *Classic of Mountains: Southern*

Chapter 3

# Basic Principles

In this chapter, we give an overview of the basic principles of the techniques we used in this thesis. Section 3.1 covers the theories of the SW. In Section 3.2 we provide the reader with the basic philosophy of RDF. Section 3.3 confers an overview of the principles of ontologies. In Section 3.4 and Section 3.5 the SPARQL query language and triplestores are described respectively. Section 3.6 introduces the concepts of Linked Data, and conclusions are made in Section 3.7.

## 3.1 Semantic Web

> *"The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."*
>
> *Tim Berners-Lee, James Hendler, Ora Lassila (2001)*

The Semantic Web (SW) [Berners-Lee et al., 2001] is a concept proposed by Tim Berners-Lee in 1998 [Yu, 2011]. It is an extension of the World Wide Web (WWW) through standards by the World Wide Web Consortium (W3C) [Tilahun and Kauppinen, 2012], rather than a separate set of distinct websites. The core idea of the SW is to add machine-readable metadata to the documents (such as HTML) on the WWW to make the WWW become a universal information exchange medium.
The Semantic Web Stack in Fig. 3.1 is the architecture of SW, which uses graphical interpretation as a hierarchy of languages at different levels. Each of these levels utilizes the downstream capabilities. It shows how technologies that are standardized for SW are organized to make the Semantic Web possible and how SW is an extension (not replacement) of classical hypertext Web [Spahiu, 2017].
In the following we will describe four of the building blocks: RDF, RDFS, OWL and SPARQL to present the way we represent the data and query the data.

[1] Adapted from https://en.wikipedia.org/wiki/Semantic_Web#/media/File:Semantic_web_stack.svg

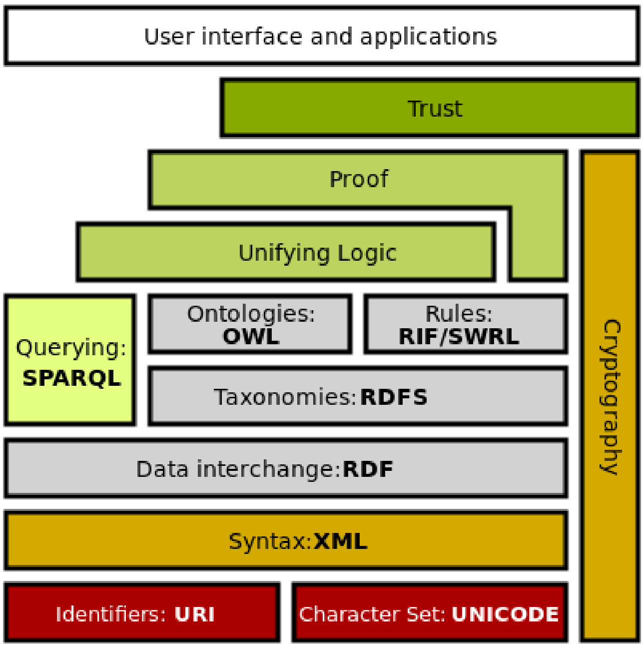


Figure 3.1: Semantic Web Stack[1]

## 3.2 Resource Description Framework

### 3.2.1 Data Model

Resource Description Framework (RDF) is a framework for representing information in the Web [Klyne et al., 2004]. At the core of RDF is a model for representing named properties and their values, which serve both to express attributes of resources (and in this sense correspond to usual attribute-value-pairs) and to present relationships between resources [Lassila and Swick, 1997]. Both the resources and attributes are shown with a URI [Coates et al., 2001].
The RDF is a simple language, based upon the idea of making statements about resources in the form of subject-predicate-object expressions known as triples [Yarandi, 2013]. The Subject denotes the resource, and the Predicate denotes the traits or aspects of the resource and expresses a relationship between the Subject and the Object [Akoumianakis, 2009].

**Example 1.** Description of a monster named "Gibbon-like (玃如)[2]".

This example is derived from our RDF triples. The original text in both English and classical Chinese versions are listed in List.1 and List.2, where different terms in the text that we extract to make up for the RDF triples are highlighted with distinct colours.

[2]This monster is from the chapter *Classic of Mountains: Western* of *Shanhaijing*

```
Three hundred and eight leagues further west is a mountain called Mount Bogmire...
There is an animal here which looks like a stag, but it has a white tail, horse's hoofs,
and human hands, and it has four horns, its name is the gibbon-like
```

Listing 1: Text of the Monster Gibbon-like in [Birrell, 1999]

```
西南三百八十里，曰皋涂之山，有兽焉，其状如鹿而白尾，马脚人手而四角，名曰玃如
```

Listing 2: Text of the Monster Gibbon-like in [Yuan, 1980]

Table 3.1 represents RDF statements of the resource Gibbon-like, which is a monster in *Shanhaijing* living in the Mount Bogmire (皋涂之山). In this table, the monster's name is illustrated in both Chinese and English, along with its properties that we construct from the terms highlighted in the raw text:

| Subject | Predicate | Object |
|---|---|---|
| chm-r:Gibbon-like | rdf:type | chm-o:monster |
| chm-r:Gibbon-like | rdfs:label | "Gibbon-like"@en |
| chm-r:Gibbon-like | rdfs:label | "玃如"@cn |
| chm-r:Gibbon-like | chm-o:footLike | chm-r:Horse |
| chm-r:Gibbon-like | chm-o:handLike | chm-r:Human |
| chm-r:Gibbon-like | chm-o:looksLike | chm-r:Stag |
| chm-r:Gibbon-like | chm-o:tailColor | chm-rl:White |
| chm-r:Gibbon-like | chm-o:livesIn | chm-r:MountBogmire |
| chm-r:Gibbon-like | chm-o:hornNumber | 4 |

Table 3.1: RDF Statements of the Monster Gibbon-like

Here URIs are abbreviated with prefixes. For instance, `rdfs` represents <http://www.w3.org/2000/01/rdf-schema#>; <http://www.w3.org/1999/02/22-rdf-syntax-ns#> is written as `rdf`. Both of them are used to identify the structure of data. We adopt <http://chinesemonster.org/ to represent URIs for the monster category of *Shanhaijing*, using `chm-r` as the prefix of <http://chinesemonster.org/resource/, which is the URI for our resources; and `chm-o` as the short format for <http://chinesemonster.org/ontology/>, which covers our ontology.
RDF data can be interpreted as a tagged directed graph. The nodes that represent resources (subjects and objects in triples) are drawn as ellipses, those that express literals are depicted as rectangles, and the arcs shows named properties (predicates) [Slama, 2017]. Therefore, the above triples could be signified as a tagged directed graph, as shown in Fig. 3.2.

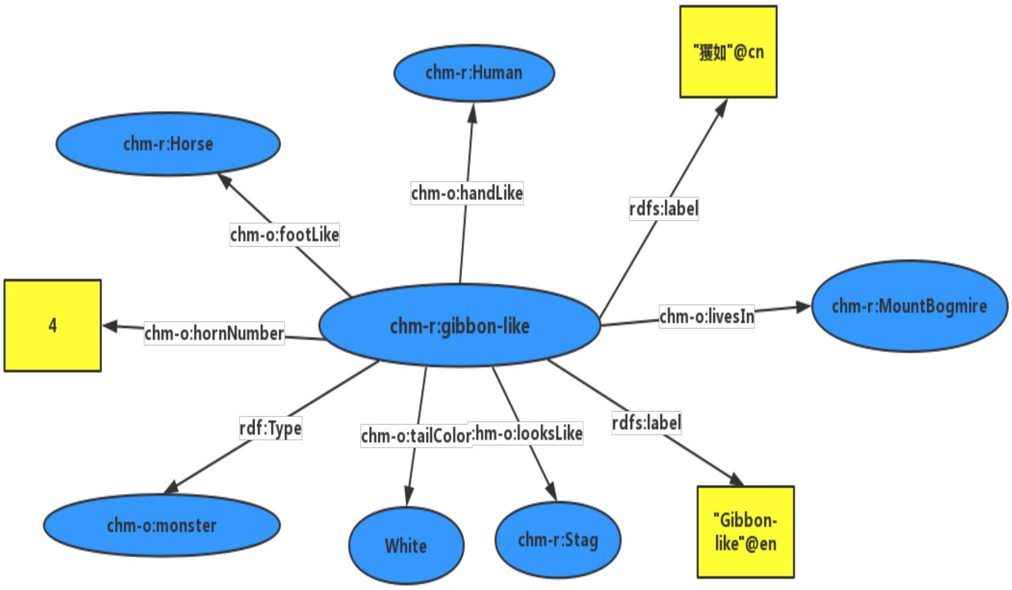


Figure 3.2: RDF Graph of the Monster Gibbon-like

Each node and predicate in RDF is identified with a URI. RDF also allows nodes that are not determined by URIs or literal strings called Blank Node or Blank Node Identifier, as temporary, internally visible identifiers for local references.

### 3.2.2 RDF Serialization Formats

Rather than a data format, RDF is a data model that describes resources in the form of a triple (subject, predicate, object), which must first be serialised using an RDF syntax to publish an RDF graph on the Web [Heath and Bizer, 2011]. This simply means taking the triples that make up an RDF graph, and using a particular syntax to write these out to a file [$T\zeta\alpha\nu\alpha\kappa\eta$, 2016]. Currently, there are several formats to serialise RDF, such as RDF/XML, N-Triples, Turtle, RDFa, and JSON-LD.

- **RDF/XML** A syntax developed by the W3C that is employed to interpret RDF data in XML format. The reason for proposing this approach is that XML is a mature technology with multiple off-the-shelf tools to store and parse XML. However, XML format for RDF is abundant and not as human-friendly as other serialization formats. List. 3 indicates the triples of the monster Gibbon-like encoded in RDF/XML format.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
   xmlns:chm-o="http://chinesemonster.org/ontology/"
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
>
  <rdf:Description
   ↪  rdf:about="http://chinesemonster.org/resource/gibbon-like">
    <chm-o:footLike
     ↪  rdf:resource="http://chinesemonster.org/resource/Horse"/>
    <chm-o:looksLike
     ↪  rdf:resource="http://chinesemonster.org/resource/Stag"/>
    <chm-o:hornNumber>4</chm-o:hornNumber>
    <rdfs:label xml:lang="cn">攫如</rdfs:label>
    <rdf:type
     ↪  rdf:resource="http://chinesemonster.org/ontology/monster"/>
    <chm-o:handLike
     ↪  rdf:resource="http://chinesemonster.org/resource/Human"/>
    <chm-o:livesIn
     ↪  rdf:resource="http://chinesemonster.org/resource/Mount
     ↪  Bogmire"/>
    <rdfs:label xml:lang="en">Gibbon-like</rdfs:label>
    <chm-o:tailColor
     ↪  rdf:resource="http://chinesemonster.org/resource/White"/>
  </rdf:Description>
</rdf:RDF>
```

Listing 3: RDF/XML Format of Triples of Monster Gibbon-like

- **N-Triples** The most intuitive representation, using multiple triples to represent RDF datasets. In a file, each row denotes a triple, which is convenient for machine parsing and processing.

- **Turtle** One of the most used RDF serialization techniques. It is more compact compared with RDF/XML and more readable than N-Triples. List 4 shows the same triples of the monster Gibbon-like encoded in Turtle format.

```
@prefix chm-o: <http://chinesemonster.org/ontology/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix chm-r: <http://chinesemonster.org/resource/> .

chm-r:Gibbon-like rdf:Type chm-o:monster ;
                  rdfs:label "Gibbon-like"@en ;
                  rdfs:label "玃如"@cn;
                  chm-o:footLike chm-r:Horse ;
                  chm-o:handLike chm-r:Human ;
                  chm-o:looksLike chm-r:Stag ;
                  chm-o:tailColor chm-r:White ;
                  chm-o:livesIn chm-r:MountBogmire ;
                  chm-o:hornNumber "4"  .
```

Listing 4: Turtle Format of the Monster Gibbon-like

- **RDFa** The Resource Description Framework in Attributes, an extension of HTML5. It allows the site builder to mark up entities like people, places, comments on the page without changing any display effects.

- **JSON-LD** JSON for Linking Data, which stores RDF Data in a key-value pair.

## 3.3 Ontology

An ontology is an explicit specification of a conceptualization [Gruber, 1993]. In computer science and information science, an ontology is a formal naming and definition of the types, properties, and interrelationships of the entities that really or fundamentally exist for a particular domain of discourse [Vasant and Voropai, 2016]. In other words, An ontology defines the basic terms and relations comprising the vocabulary of a topic area, as well as the rules for combining terms and relations to define extensions to the vocabulary [Neches et al., 1991]. Fig.3.3 is a habitat ontology example edited on our *Shanhaijing* ontology. Based on Fig.3.3, the following explains four keywords of the ontology concept.

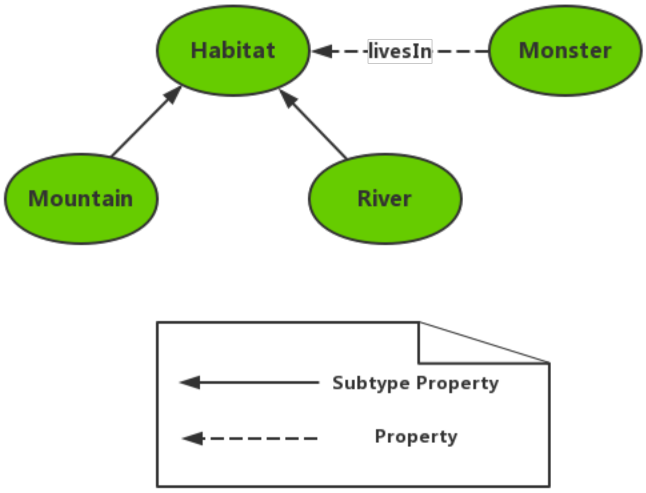


Figure 3.3: A Habitat Ontology Example Extracted from the *Shanhaijing* Ontology

- **Domain of Discourse** An ontology describes a specific domain. For example, in Fig. 3.3 we decide that the area to be specified this time is "habitat".
- **Term** An ontology uses terms to refer to essential concepts in a given domain. For example, to determine what is vital to the habitat, we can enumerate the concepts of "habitat", "mountain" and so on.
- **Relationships** The relationships of terms are expressed as a hierarchy, which defines the interactions among these terms. For example, in Fig. 3.3, The term `Mountain` and `River` are subtypes of `Habitat`. Besides, the terms `Monster` and `Habitat` interact with each other with a property `livesIn` rather than the subtype property.
- **Rules** Properties determine rules to constraint the relationships between subject and objects. For example, only the `Monster` could lives in `Habitat`, the term `River` is not allowed to overlap with term `Mountain`.

### 3.3.1 Ontology Languages

An ontology language is a formal language used to encode the ontology [Maniraj and Sivakumar, 2010]. There are several such kinds of languages for specifying interactions of terms of the ontology. Here two of them (RDF Schema and the Web Ontology Language) are introduced.

#### 3.3.1.1 RDF Schema

RDF Schema is the most basic schema language which provides declarative schemata whose semantics are defined within RDF Schema [Ashraf, 2013]. It takes <http:

//www.w3.org/2000/01/rdf-schema#> as a namespace vocabulary on the basis of RDF, as a standard for describing classes and attributes in a particular domain. In this thesis, we use the prefix `rdfs:` to represent <http://www.w-3.org/2000/01/rdf-schema#> and the prefix `rdf:` for <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
In RDF, a class is an abstraction of a set of individual resources, each of which is referred to as an instance of a class [Alam, 2015]. For example, `Monster` is a class, and the specific monster `Gibbon-like` is an instance of `Monster`. `rdfs:Class` is a class of all classes, `rdfs:Resource` is a class of all resources. `rdfs:subClassOf` and `rdfs:subPropertyOf` are adopted to represent the inheritance relations between classes and properties respectively. For example, `Mountain` and `River` are subclasses of class `Habitat`. While `rdfs:domain` and `rdfs:range` are employed to describe the domain and range of attributes. Besides, we prefer `rdf:type` to express a resource belonging to a class, and `rdf:Property` denotes a class of all properties.

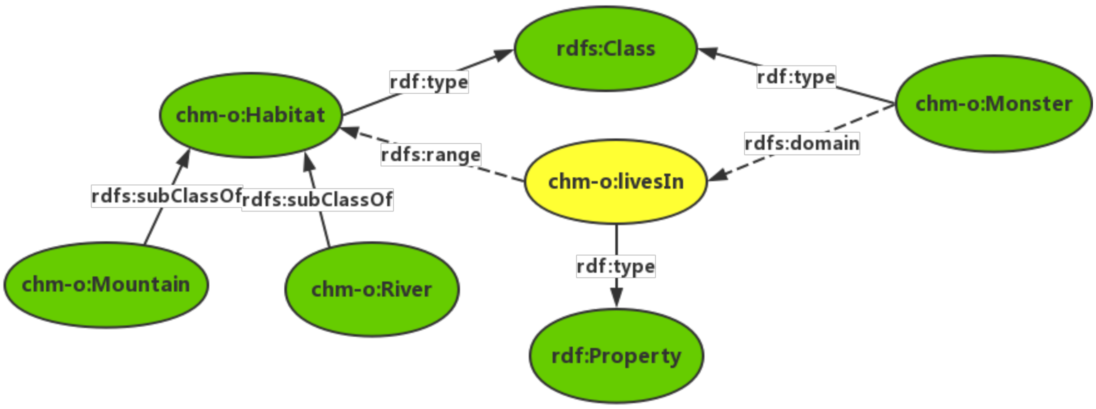


Figure 3.4: An RDF Schema Example of the Habitat Ontology Extracted from *Shanhaijing*

Fig.3.4 depicts an example of using RDF Schema to form the ontology. In Fig.3.4, the class `Habitat` and `Monster` are two instances of `rdfs:Class`, the property `livesIn` is an instance of `rdf:Property`, with a domain `Monster` and a range `Habitat`. While classes `Mountain` and `River` are subclasses of `Habitat`.
The RDF representation of the RDF Schema example shown in Fig. 3.4 is given in List.10 in Turtle serialization.

```
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix chm-o:  <http://chinesemonster.org/ontology/> .

chm-o:Habitat a rdfs:Class .

chm-o:Mountain a rdfs:Class ;
               rdfs:subClassOf chm-o:Habitat .

chm-o:River a rdfs:Class ;
            rdfs:subClassOf chm-o:Habitat .

chm-o:Monster a rdfs:Class .

chm-o:livesIn a rdfs:Property ;
         rdfs:domain chm-o:Monster ;
         rdfs:range  chm-o:Habitat .
```

Listing 5: An RDF Representation of the *Shanhaijing* Habitat Ontology in Turtle

#### 3.3.1.2 Web Ontology Language

The Web Ontology Language (OWL), built on RDFS, is used to create ontologies and is also a W3C recommendation [Bechhofer et al., 2004]. OWL has two main functions: providing fast, flexible data modeling capabilities as well as offering efficient automatic reasoning.

OWL is abundant in expressive power with its various vocabularies. Some vocabularies in OWL are defined for describing attribute characteristics. For example, `owl:TransitiveProperty` indicates that the attribute has a transitive property. `owl:SymmetricProperty` represents the property has symmetry. `owl:FunctionalProperty` denotes the uniqueness of the value of this property. `owl:inverseOf` defines the inverse relationship of an attribute. We take `owl:inverseOf` as an instance. In our ontology, we define the property `childOf` as an inverse property of `hasChild`, which means, for instance, the monster `Weeape (季禺)`[3] is a child of `FondCare (颛顼)`, then `FondCare` must has a child of `Weeape`.

Despite that, many vocabularies are employed for ontology mapping. For example, `owl:equivalentClass` indicates that a class is same as another class. Similarly, `owl:equivalentProperty` denotes a property is equivalent to another property, and `owl:sameAs` represents that two entities are considered as the same. Ontology map-

[3]This monster comes from the chapter *Classic: the Great Wilderness: Southern* of *Shanhaijing*

ping is mainly used to integrate multiple independent ontologies. One instance is that, as shown in List.11, we construct an ontology where a class `Monster` is utilised to describe those unimaginable animals that with various characteristics in mythologies, while in WikiData, another class `MythicalCreature` represents creatures in the same domain. So we could identify that these two classes share a similar meaning and could be interlinked together. When we merge these two ontologies, we can apply the OWL mapping vocabulary.

```
<http://www.chinesemonster.org/ontology/monster> a owl:Class ;
   owl:equivalentClass <https://www.wikidata.org/wiki/Q2239243>.

<https://www.wikidata.org/wiki/Q2239243> a owl:Class .
```

Listing 6: Ontology Mapping Example of Class `Monster` with OWL

The knowledge graph reasoning is divided into two categories: ontology-based reasoning and rule-based reasoning. Here we introduce the ontology-based reasoning. As mentioned above, OWL has the capability of practical automatic reasoning. For example, our database consists of kinship relationships between monsters, many of which are one-way, such as our dataset only saves that `Weeape` is the child of `FondCare`, but `Weeape` is not included in the children field of `FondCare`. Now we apply `owl:inverseOf` for the properties `childOf` and `hasChild`, by which relationships between these two monsters are completed. As Fig. 3.5 shows, the solid line represents the relationship existent in the database, while the dashed line shows the association inferred by OWL.

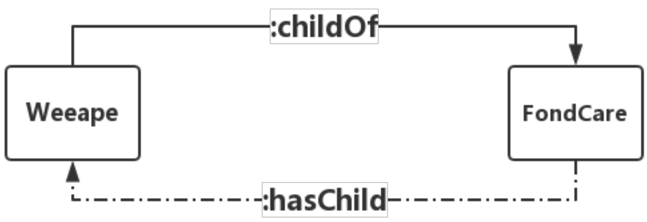


Figure 3.5: OWL Example of InverseOf Properties `childOf` and `hasChild`

### 3.3.2 URIs Designing

LD resources are identified by URIs. It is an important step in any LD project to define the conventions for URI assignments [Szász et al., 2016]. Here we adopt the Cool URI [Berners-Lee, 1998], which is used to describe the term that does not change.

The URIs should be designed with simplicity, stability and manageability in mind, thinking of them as identifiers rather than as names for Web resources [Pan et al., 2017]. Here are some guidelines to design URIs outlined by Villazón-Terrazas et al. [2011]:

- Use meaningful URIs, instead of opaque URIs, when possible.
- Use slash (303) URIs, instead of hash URIs, when possible.
- Separate the TBox (ontology model) from the ABox (instances) URIs.

### 3.3.3 Ontology Reuse Rules

The reuse of existing vocabularies is encouraged as much as possible, which links different resources from the same domain together, thus improving the ontology development and assuring interoperability. Followed and summarised from Villazon-Terrazas's guideline paper [Villazón-Terrazas et al., 2011], three tasks need to be taken for this activity:

- Search for suitable vocabularies to reuse.
- If no available vocabulary suits our domain, we need to create one with reusing existing resources as much as we can.
- If no vocabularies or resources are acceptable for us, we need to create them from the scratch.

The reuse of ontologies is critical to their value as a means of knowledge representation [Katsumi and Grüninger, 2016]. Ontology reuse involves building a new ontology through maximising the adoption of pre-used ontologies or ontology components [Lonsdale et al., 2010]. Advantages of ontologies reuse consist of reducing the labour of building ontologies from scratch, as well as increasing the quality of new ontologies because the reused components have already been tested [Lonsdale et al., 2010]. If a new ontology has to be created, conservative and gradual approaches are recommended.

### 3.3.4 Ontology Building Approaches

There are many accessible ontology building tools which could be used for creating, editing and manipulating ontologies, such as the NeOn toolkit [Haase et al., 2008], Protege and Knoodl [Fu and Rao, 2015], etc.
To analyse the collected terms, we use logic order to merge similar terms and organise relevant terms into a taxonomic (subclass) hierarchy. Regarding establishing classes and determine the class hierarchy, three major approaches, top-down, bottom-up and the combination of both are widely applied.
The top-down approach development process begins with the ideas of the most general concepts in the domain and subsequent specialisation of the concepts, and the

bottom-up approach development process starts with the definition of the most specific classes, the leaves of the hierarchy, with a subsequent grouping of these classes into more general concepts [Noy et al., 2001]. The combination process is the mixture of the two approaches, by which we could generate a few top-level concepts and a few specific classes, then relate them to a middle-level concept.
The classes alone are not able to express enough information of the domain. After defining classes, the internal structure of concepts is explained, where we employ properties. Properties represent relationships between individuals, which link the individual from the domain to the individual from the range. There are three different types of properties: (1) object properties, (2) datatype properties, and (3) annotation properties.
The object properties link individuals to other individuals. Datatype properties relate individuals to XML schema datatype values or the RDF literal data, and Annotation properties are often used to add information to classes, individuals, object properties and datatype properties [Horridge et al., 2004].

## 3.4 SPARQL Query Language

SPARQL (SPARQL Protocol and RDF Query Language), is a query language and a data access protocol developed for RDF [Clark et al., 2006]. It is defined for the RDF data model produced by the W3C but can work for any information resource that represented in RDF. For open datasets, agents use a SPARQL endpoint to access the data with SPARQL queries. SPARQL presents four varying types of queries written in Turtle format:

- **SELECT** In the query matching mode returns all or part of the subset.
- **CONSTRUCT** Returns a new RDF graph pattern by replacing the variables in the triples.
- **ASK** Returns a boolean value indicating whether there is a match.
- **DESCRIBE** Returns an RDF graph schema that represents the resource.

Each of these query forms takes a WHERE block to restrict the query, except that in the case of the DESCRIBE query, the WHERE is optional [Qi et al., 2013].
To see the monster whose tail color is white in *Shanhaijing*, the SPARQL query (shown in List.7) is utilised, where `monster` is a variable indicated by a "?".

```
PREFIX chm-o: <http://www.chinesemonster.org/ontology/>
PREFIX chm-r: <http://www.chinesemonster.org/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?monster
WHERE {?monster rdf:type chm-o:monster .
       ?monster chm-o:tailColor chm-r:white .
       }
```

Listing 7: SPARQL Query for Monster's Tail Color

## 3.5 Triplestore

Triplestores or RDF stores are purpose-built databases for the storage and retrieval of any type of data expressed in RDF [Modoni et al., 2014]. Triplestores enable users to execute queries more smoothly and efficiently. There are several triplestores available such as Virtuoso [Idenhen, 2013], which is often used in academic projects. Others like GraphDB [Güting, 1994], AllegroGraph [Aasman, 2006] and Apache Jena [Jena, 2007], all of which are widely employed and powerful databases for RDF. In this thesis, we adopt Apache Jena for the storage of our data.

## 3.6 Linked Data

As the "Linked Data FAQ" of Structured Dynamics states, "LD is a set of best practices for publishing and deploying instance and class data using the RDF data model, naming the data objects using uniform resource identifiers (URIs), thereby exposing the data for access via the HTTP protocol, while emphasizing data interconnections, interrelationships and context useful to both humans and machine agents" [Complicated, 2009]. LD creates links between disparate data sources, which can be separate databases maintained by two institutions in different geographic locations, or may be distinct systems within the same organisation that cannot interact at the data level [Gao, 2012]. LD refers to the data published on the Web, with an explicit meaning and machine readability, connecting to external datasets as well as being linked by other outer databases. Different from current hypertext network, the basic unit of which is the HTML (hypertext markup language) file connected by a hyperlink, LD uses RDF to form a data network that links everything in the world rather than merely connects these files, thus being an extension of the hypertext network.

The adoption of LD over the last few years has raised from 12 datasets in 2007, to more than 1000 datasets as of April 2014 [Schmachtenberg et al., 2014], and in 2017, the number has significantly increased as the Fig.3.6 shown. Fig.3.6 is the LD Cloud

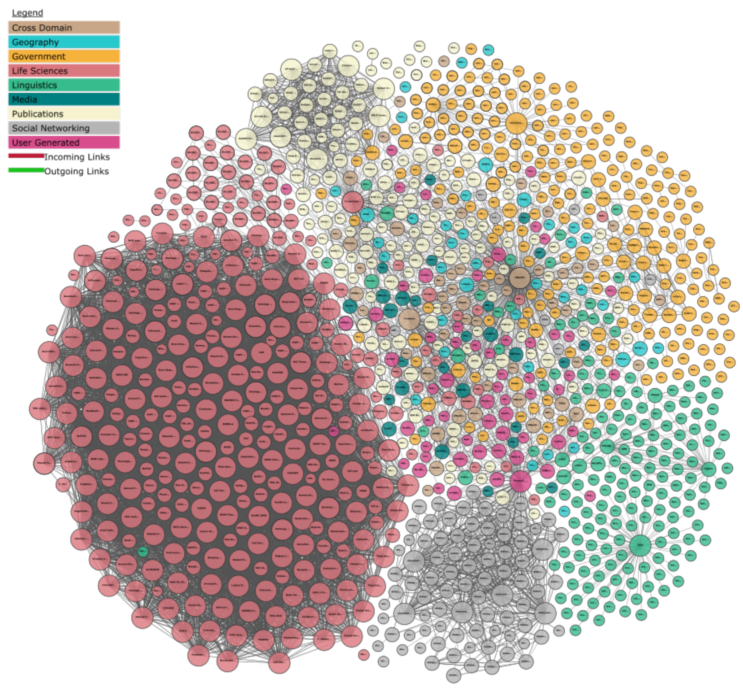


Figure 3.6: Linked Open Data Cloud Diagram [Abele et al., 2017]

Diagram under an open license, where different colours represent different domains.

Four principles for publishing LD on the Web have been defined by Berners-Lee [2006]:

- Use URIs as names for things
- Use HTTP URIs so that people can look up those names
- When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL)
- Include links to other URIs, so that they can discover more things.

The third and fourth principles require that RDF files contain useful information and as many URIs as possible, thus expecting LD RDF files involving nearly no blank nodes and few literals. In this case, the `blank node` is a local resource with no global ID (the URI of the named domain is not defined), such as ISBN, DOI, and the `literal` refers to a string value that can have type and language properties. Both of them cannot be employed to point to a `resource`. Hence the massive usage of them fails to link the data as expected.
In 2010, Tim Berners-Lee has proposed a 5-star ranking system to evaluate the quality and availability of Open Data. Closely related to the four principles above, the 5-star ranking system is as follows [Berners-Lee, 2006]:

- ☆Available on the web (whatever format) but with an open license, to be Open Data
- ☆☆Available as machine-readable structured data (e.g. excel instead of image scan of a table)
- ☆☆☆Available as machine-readable structured data with non-proprietary format (e.g. CSV instead of excel)
- ☆☆☆☆Use open standards from W3C (RDF and SPARQL) to identify things, so that people can point at your stuff
- ☆☆☆☆☆Link your data to other people's data to provide context

Additionally, even in the Linked Open Data Cloud diagram (Fig.3.6), most publishers do not publish their data with an explicit license [Abele et al., 2017]. Besides, the techniques developed for open datasets are applicable for all linked datasets. Moreover, as Tim Berners-Lee mentioned, any source that satisfies the 5-star schema is considered as a 5-star LD, no matter it is open or not. Hence in this thesis, we treat the Linked Open Data same as LD. Despite that, our data meets the four Linked Data principles as well as the 5-star ranking schema.

## 3.7 Conclusion

In this chapter, we introduced the basic principles and related techniques that support this thesis, along with ontology and data examples from our work. We introduced theories of the SW in Section 3.1, while Section 3.2 covers an overview of RDF, consisting of the RDF data model and its serialization formats. In Section 3.3, we described the concept of Ontologies, where two essential ontology languages, RDFSchema and OWL are included as well. We also contains some basic rules for building an ontology Section 3.4 and Section 3.5 involves the use of SPARQL query language and a brief presentation of triplestores respectively. In Section 3.6, we talked about Linked Data, which is an important component of this thesis. Four principles of publishing LD and the five-star LD ranking system are shown in this section.

Chapter 4

# Constructing the Shanhaijing Ontology and LD

In this chapter, we describe the process of building the ontology for monsters in *Shanhaijing* and the means of managing the data. This chapter introduces the standards we used for producing our ontology and data, which includes Specification, Modeling, Generation, Publication and Exploration parts. In Section 4.7, we conclude contents of this chapter.

## 4.1 Preliminary

An ontology is the collection of interrelated semantic-based modeled concepts on the basis of already defined finite sets of terms and concepts used in information integration and knowledge management [Ahmed and Gerhard, 2010]. Ontologies form the backbone of the LD environment and are key to supporting its open and distributed infrastructure [Dutta, 2017]. In 2011, Villazón-Terrazas et al. provided a 5-stage methodological guideline for publishing government LD, which consists of the following main activities: (1) specification, (2) modeling, (3) generation, (4) publication, and (5) exploitation [Villazón-Terrazas et al., 2011]. In this thesis, we accept the five activities as guidelines to demonstrate the process of building the ontology for the monsters of *Shanhaijing*.

## 4.2 Specification

During the specification stage, traditional ontology engineering requires the selection of the appropriate level of formality for coding [Gomez-Perez et al., 2004].

### 4.2.1 Scope and Purpose

The specification phase in the development of applied ontologies focuses on the definition of the purpose and the scope of an ontology [Pattuelli et al., 2015]. In this thesis, rather than concentrate on the contents of the whole book, we capture the data of all of the monsters in *Shanhaijing* with both the classical Chinese version and the English transcript, leaving the complete contents as a follow-up in the future. Reasons for focusing on this domain are as follows:

- **Content Description** Compared with other aspects such as geography, witchcraft, plants, and worship ceremonies, the length of the content used to depict the monsters is the longest. Besides, the fascinating and detailed account of the monster category overwhelms the other aspects.
- **Influence** No matter which parts of the Chinese literature or other countries' mythologies inspired from *Shanhaijing*, the derivations of monsters continuously account for a notable proportion.
- **Reuse Current Folktale Ontologies** Compared with other aspects, part of current folktale ontologies that created for Western myths could be reused relatively more adequately for the monsters. Also, after our ontology completed, links to other mythic ontologies is forward to be more accessible.

The primary goal of our ontology is by utilising LD principles and technologies to present the characteristics of the mythic monsters in *Shanhaijing* and the relationships among them, and link to and enrich current mythological ontologies. In doing so, we are providing an efficient way for users who are interested in Chinese myths to carry out explorations, analyses, and research into *Shanhaijing* and contributing to the richness of folktale ontologies.

## 4.3 Formalization

Following the guidelines in Chapter 3.3.2, our URIs are defined under the pattern of: http://chinesemonster.org/ontology for the underlying URI of the ontological model and http://chinesemonster.org/resource for the resource primary URI. For example, to denote a specific monster `Nine-tailed Fox`, the following URI http://chinesemonster.org/resource/nine-tailed_fox is utilised.

## 4.4 Modeling

After the specification activity, in which the data sources were identified, selected and analyzed, we need to determine the ontology to be used for modeling the domain of these data sources [Pan et al., 2017].

### 4.4.1 Existing Ontologies Reuse

In this thesis, we carried out a large number of document retrieval in separate ontology libraries to see if any mythological ontologies are available for our data (see Table 4.1).

Only a few mythological ontologies are reusable for this thesis, one of which is the

| Library | URL |
|---|---|
| Swoogle | http://swoogle.umbc.edu/2006/ |
| I3CON | http://www.atl.lmco.com/projects/ontology/i3con.html#Ontologies |
| DAML Ontology Library | http://www.daml.org/ontologies/ |
| Protege Ontology Library | https://protegewiki.stanford.edu/wiki/Protege_Ontology_Library |
| LOV | https://lov.okfn.org/dataset/lov |
| ONKI | https://onki.fi/en/ |
| The OBO Foundry | http://owl.cs.manchester.ac.uk/repository/ |

Table 4.1: Ontology Libraries Used for Mythological Ontologies Retrieval

Mahabarata Ontology [Protege Wiki, 2018], which is based on one of the epic stories of ancient India Mahabarata.
Ontologies describing animals, their body parts and their habitats could also be adopted since our purpose was to demonstrate the characteristics of monsters of *Shanhaijing*, which share some similarities with real animals. DBpedia Ontology [DBpedia, 2014] is a general source of semantics which describes common entities. Domain-specific ontologies representing animals and habitats, such as BioTopOntology [Whetzel et al., 2011] were also considered.
In addition to these domain-specific ontologies, we also apply common vocabularies for expressive predicates to process the content of information, such as RDFS, OWL, SKOS, XML Schema, RDF, etc. Table.4.2 demonstrates most external vocabularies that we utilise for our linked data.

| Vocabulary-Prefix | Namespace |
|---|---|
| dbpedia-owl | http://dbpedia.org/ontology/ |
| rdf | http://www.w3.org/1999/02/22-rdf-syntax-ns# |
| rdfs | http://www.w3.org/2000/01/rdf-schema# |
| owl | http://www.w3.org/2002/07/owl# |
| skos | http://www.w3.org/2004/02/skos/core# |
| xsd | http://www.w3.org/2001/XMLSchema# |
| biotop | http://purl.org/biotop/biotop.owl# |
| mao | https://sites.google.com/site/ontoworks/ontologies |

Table 4.2: External Vocabularies Utilised for Shanhaijing Ontology

### 4.4.2 Ontology Building

As mentioned in the previous section, the vocabularies and resources we found available for our domain are limited, which means building a new ontology based on partially reusing existing sources is an ideal choice for this thesis. Steps for creating our ontology are (1) enumerate terms; (2) define taxonomy; (3) define properties.

#### 4.4.2.1 Enumerate Terms

After specifying the domain of this thesis, we rely on traditional humanities methodologies to do the close reading of the resources in both Chinese and English. We identify all of the relevant terms and record them in an unstructured list. The terms we generated contain body parts descriptions, characteristics, as well as habitats which are related to monsters. List.8 is an example of the Nine-tailed fox[1] to showcase how we extract terms from the text.

```
Three hundred leagues further east is a mountain called Mount Greenmound...
There is an animal on its mouontain that looks like a fox, but it has nine tails.
It makes a noise like a baby. It can devour humans. Whoever eats it will not
be affected by malign forces.
```

Listing 8: Text of the Monster Nine-tailed Fox in English [Birrell, 1999]

```
又东三百里, 曰青丘之山...有兽焉, 其状如狐而九尾. 其音如婴儿. 能食人. 食者不蛊.
```

Listing 9: Text of the Monster Nine-tailed Fox in classical Chinese [Yuan, 1980]

List.8 contains a piece of text excerpted from the English version of *Shanhaijing*, which introduces the monster `Nine-tailed Fox`. The related classical Chinese text is shown in List.9. We consider the two translations at the same time. In order to precisely and clearly describe terms in this book, we provide names of entities in both languages at the first time they appear in this thesis, and adopt English translations for further presentations. During the deep reading process, we extract all of the concepts related to our domain, which are highlighted in this example, then split them into terms. In this example, terms `monster`, `mountain`, `location`, `tail`, `distanceFrom`, `directionFrom`, `livesIn`, `looksLike`, `tailNumber`, `noiseLike`, `die-taryCharacter`, `efficacyByEaten` are summarized and listed as the basis of classes and properties of our ontology.
Due to the lack of an available dataset containing structured data of the contents of *Shanhaijing*, while extracting the terms, we create a dataset categorizing both the character types, and the types of relationships between them at the meantime. In this example, the information `mountain:Mount_Greenmound(青丘之山)`, `distanceFrom:300`, `directionFrom:east`, `monster:Nine-tailed_Fox`, `livesIn:Mount_Greenmound`, `tailNumber:9`, `looksLike:fox`, `noiseLike:baby`, `dietaryCharacter:human`, `efficacyByEaten:"against malign forces"` are stored in the dataset.

#### 4.4.2.2 Define Taxonomy

In this step, analyses of terms collected at the previous stage are employed. We select the combination approach (the mixture of top-down and bottom-up approaches) to

[1]This monster is from the chapter *Classic of Mountains: Southern*

identify the class hierarchy. Additionally, the RDF Schema vocabulary `rdfs:subClassOf` is adopted to identify the relationships between the superclass and the subclasses, which means the set of individuals in the subclass extension should be a subset of individuals in the superclass extension [Bechhofer et al., 2004].
In our ontology, a term is considered as a class if it has attributes pointing to other classes or literals, or it is the superclass of other classes. Otherwise, it is changed to a property. For example, a term `monster` is a class based on the reason that it has attributes linking to other classes, like `monster livesIn mountain`. However, the term `noise` cannot become a class because it is the domain of no attribute. Hence it is changed to a property `hasSameNoiseAs` with the domain of monster and pointing to a literal.
Based on the principles above, we create classes for our *Shanhaijing* ontology. These classes with hierarchy are shown in Fig.4.1).

#### 4.4.2.3 Define Properties

There are three different types of properties in our ontology: (1) object properties, (2) datatype properties, and (3) annotation properties, all of which are introduced below. We use `rdfs:subPropertyOf` to determine the property hierarchy.

- **Object Properties** We use object properties to link entities in our dataset. Our ontology involves the inverse property and the functional property.
  The inverse property is interpreted as `owl:inverseOf`, which is used to recognise an inverse relation between two properties. In our ontology, the properties `childOf` and `hasChild` are a pair of inverse properties, which means that if monster A has a child monster B, then monster B must be the child of monster A, and vice versa. The representation of the inverse properties `chm-o:hasChild` and `chm-o:childOf` is shown in List.10 in Turtle format:

```
chm-o:hasChild rdf:type owl:ObjectProperty ;
                owl:inverseOf chm-o:childOf .
```

Listing 10: Inverse Property Sample `hasChild` and `childOf` in *Shanhaijing* Ontology

  The functional property (expressed as `owl:FunctionalProperty`) is a property that allows one unique value for each instance. In the content of *Shanhaijing*, one monster lives in a specific habitat, which indicates that the property `livesIn` is a functional property by which one monster instance only links to one particular habitat value. (see List.11).

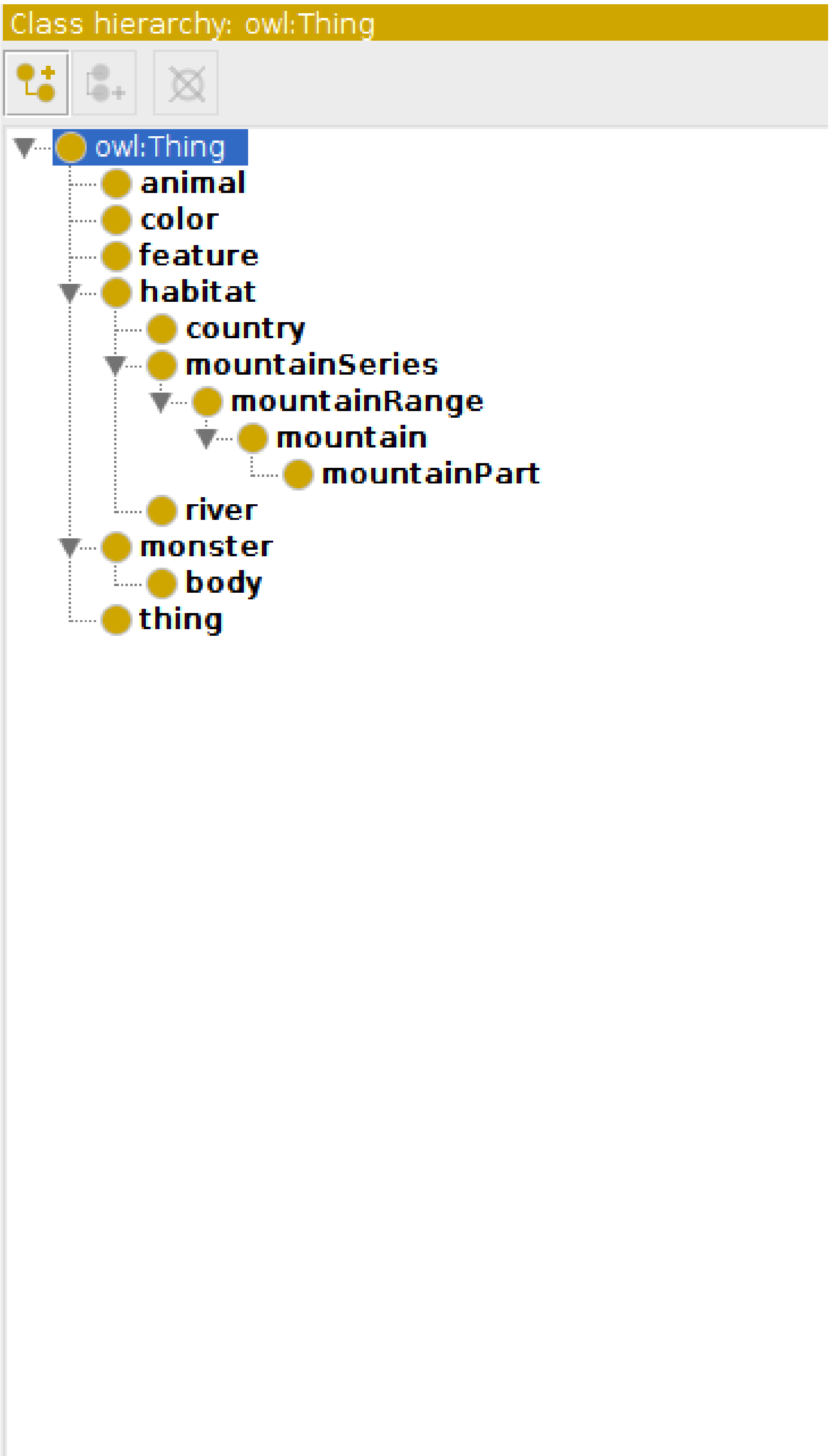


Figure 4.1: The Hierarchy of the Classes of *Shanhaijing* Ontology

```
chm-o:livesIn rdf:type owl:ObjectProperty ,
                    owl:FunctionalProperty ;
            rdfs:domain chm-o:monster ;
            rdfs:range chm-o:habitat .
```

Listing 11: Functional Property Sample `livesIn` in *Shanhaijing* Ontology

Fig. 4.2, 4.3 are figures of our object properties, where `rdfs:subPropertyOf` is employed to identify the property hierarchy. Here some important properties are defined, such as `color`, `isLike`, `isOn`, and `sizeThan`. `Color` represents the color of a monster's body parts, such as the tail's color and the eye's color, while the general concept of a monster's body parts looking like is expressed as `isLike`. The size comparison of the monster's body parts (for example the monster's eyes are larger than the eye size of a specific animal) and the unusual part the monster's body parts placed on (like the monster's wings are on its head) are defined as `sizeThan` and `isOn` respectively. Other properties such as `livesIn`, `locateAt` and `originsFrom` are used to present the information of the habitat.

- **Datatype Property** In our ontology, almost all characteristics of the monster are identified as datatype properties. One example is `hasFlavourOf`, which represents diverse flavors of monsters with the domain `monster` and the range `string`, the same syntactic construct example is shown in List. 12. Some properties utilized to describe the body part features of a monster are also treated as datatype properties, like the `eyeNumber` property, which is used to depict the number of the monster's eyes, with the domain `eye` and the range `integer`. The full version of datatype properties in our ontology is shown in Fig.4.4 and Fig.4.5, where all characteristics and some features of body parts are covered, the hierarchy of properties is also drawn.

```
chm:hasFlavourOf rdf:type    owl:DatatypeProperty ;
                 rdfs:domain chm:monster ;
                 rdfs:range  xsd:string .
```

Listing 12: Datatype Property Sample `hasFlavourOf` in *Shanhaijing* Ontology

- **Annotation Property** We use some typical annotation properties in our on-

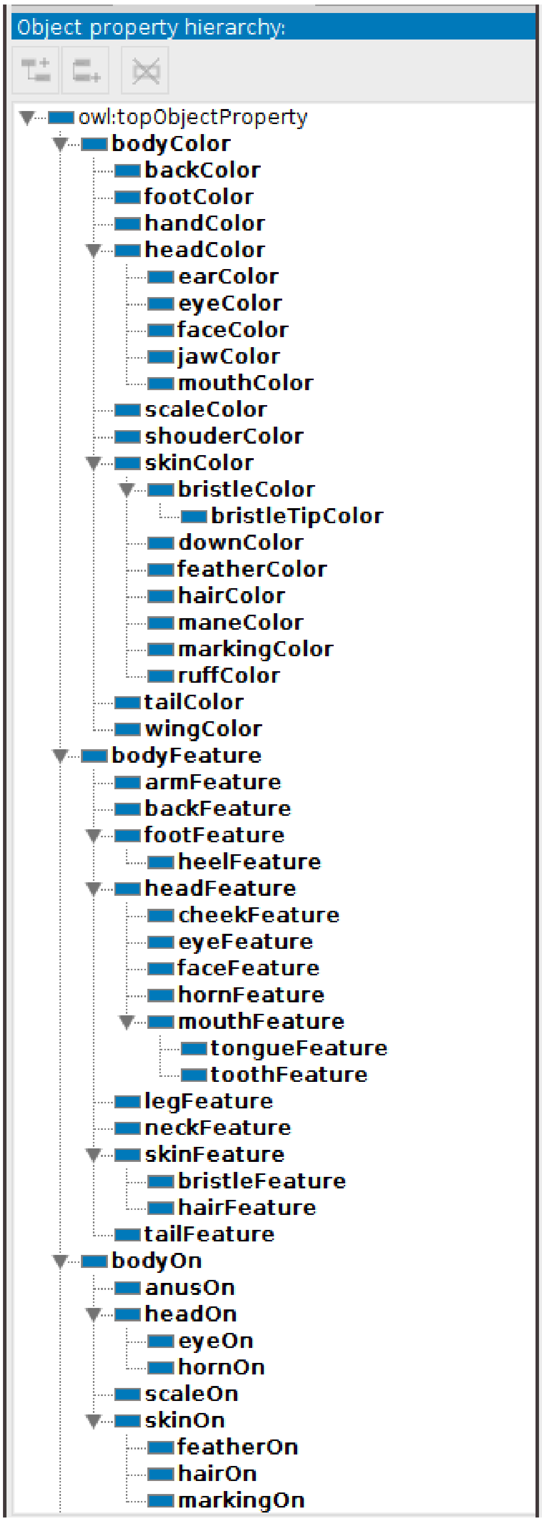



Figure 4.2: Object Properties in *Shanhaijing* Ontology: Part 1

Figure 4.3: Object Properties in *Shanhaijing* Ontology: Part 2

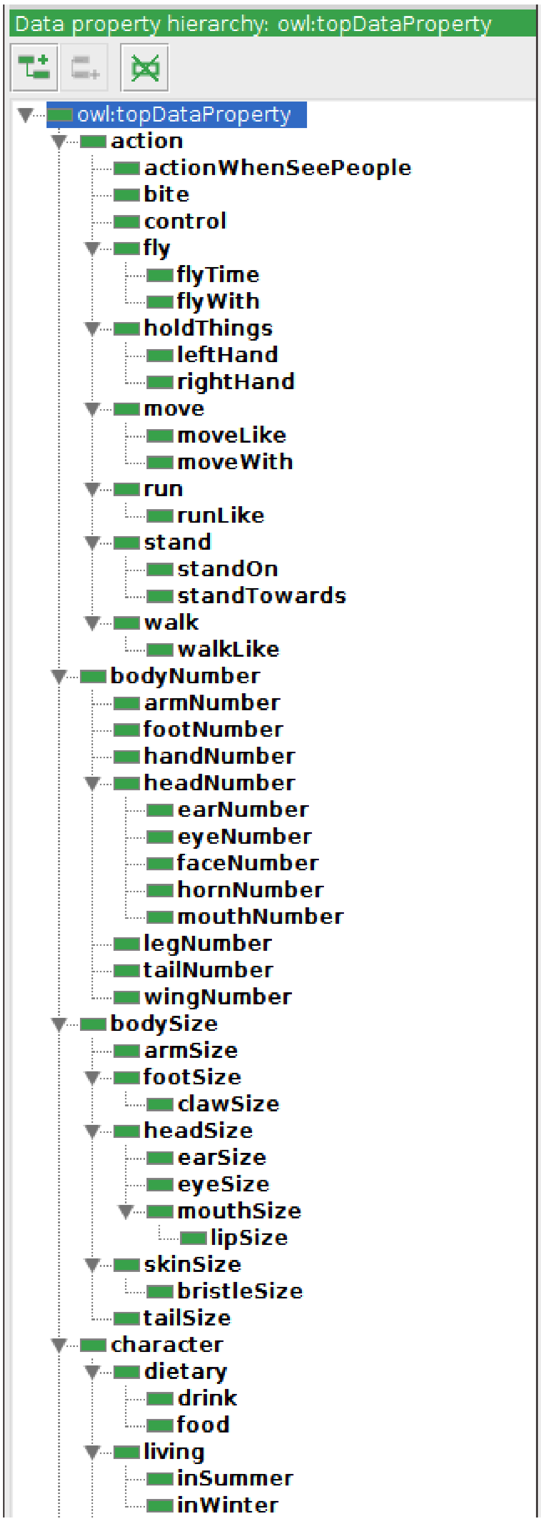


Figure 4.4: Datatype Property Structure in *Shanhaijing* Ontology: Part 1


Figure 4.5: Datatype Property Structure in *Shanhaijing* Ontology: Part 2

tology, including `rdfs:label`, `rdfs:comment` and `rdfs:seeAlso`, which are important for explaining the meaning of our ontology and make our data more human-readable. One sample of applying annotation properties is shown in List.13 , where `rdfs:label` represents the human-readable name of the class `mountain` in English language, and `rdfs:comment` explains the reason in detail why this class exists, which is "the mountain that a monster lives in".

```
chm-o:mountain rdf:type owl:Class ;
             rdfs:subClassOf chm-o:mountainRange ;
             rdfs:label  "Mountain" ;
             rdfs:comment "represents the mountain that a
 ↪  monster lives in" .
```

Listing 13: Annotation Property Sample `Mountain`

Fig.4.6 demonstrates the graphic ontology using WebVOWL [**?**], where the class hierarchy, relationships between classes, and different types of properties are contained.

## 4.5 Generation

RDF is the standard data model for data interchange and the format for our knowledge graph [Pan et al., 2017]. Therefore, in this phase, we convert data sources into RDF triples based on the vocabulary created in the modeling activity, then generate links between our database and external datasets [Pattuelli et al., 2015].

### 4.5.1 Transformation

In this thesis, the transformation involves converting data sources into RDF. As mentioned at the beginning of this chapter, to gain a better understanding and analysis of *Shanhaijing*, we select information from various data sources. Unfortunately, most of them are in different syntactic formats, which implies that converting them into RDF as well as fitting in with our ontology would be particularly hard. To overcome these difficulties, the information we adopted from different data sources needs to be analysed, preprocessed and normalised. In the modelling step we have noticed that because of the lack of appropriate ontologies and structured datasets, we have to extract them from those data sources we determined to use. Hence while doing the broad reading, not only do we enumerate terms which are related to our domain, we also excerpt instances and values corresponding to those terms and save them in our dataset. After that, we normalise our data in the dataset into a relatively structured

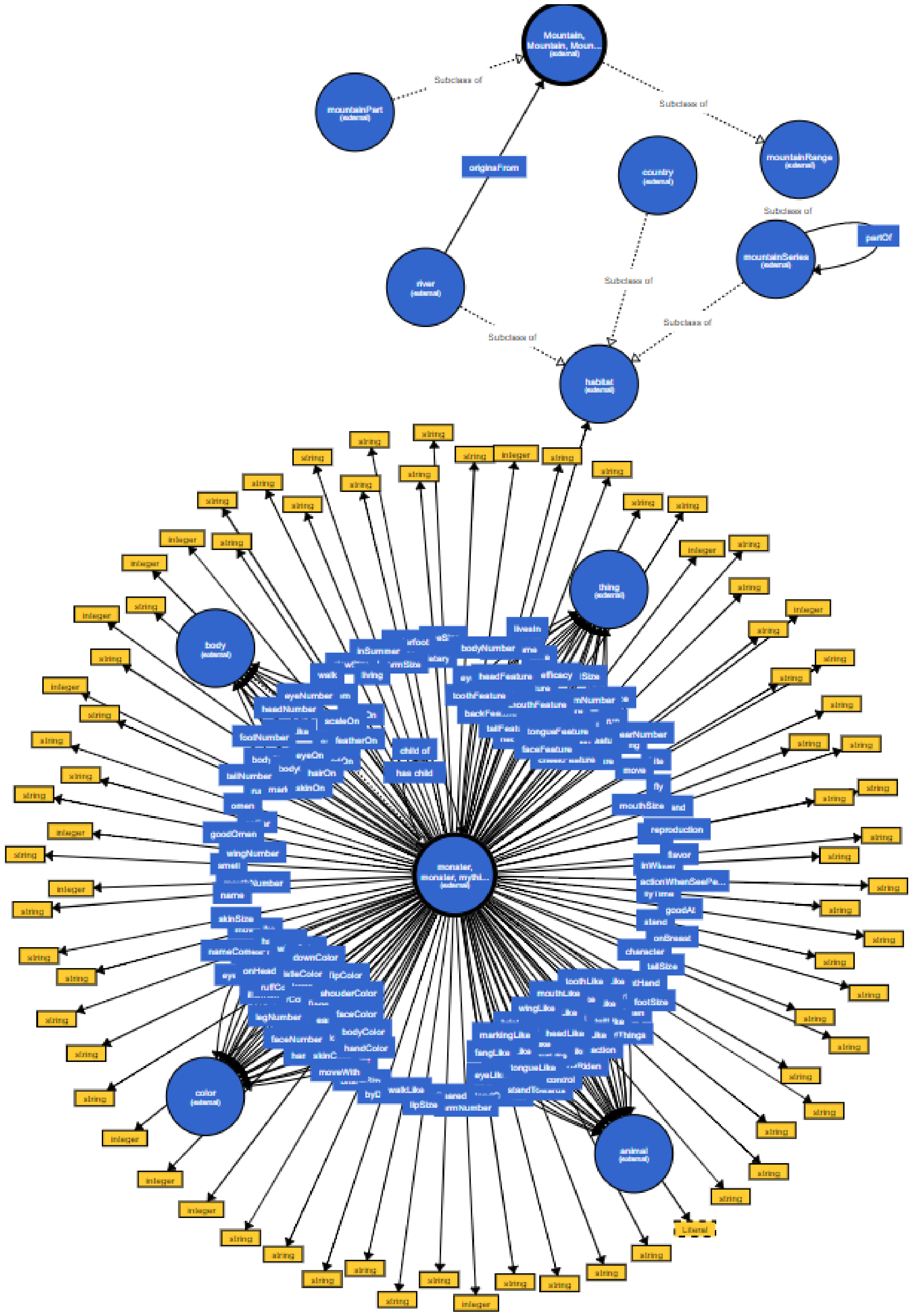

Figure 4.6: The Graphic Version of *Shanhaijing* Ontology

format that conforms our ontology, rather than in plain text. Besides, data selection, disambiguation and deduplication are contained in the preprocessing step as well. After completing the data preprocessing, we map our data with the ontology we created in the modelling activity and convert them into RDF triples. In this thesis, since our ontology is created in Protege, we adopt the MappingMaster library [O'connor et al., 2010] in Protege for the data mapping, which is an open source library that can be used to transform the content of spreadsheets to OWL ontologies. Fig.4.7 shows an example of our data mapping results, which describes features of the monster `Longright (长右)`[2]. In this example, we could recognise that *Longright* is a monster that lives in *Mount Hallcourt (柜山)*. It looks like the long-tailed ape but has four ears. Its noise is same as crooning. When it appears, the region will have significant floods. The syntactic structure of this example is presented in List.14.

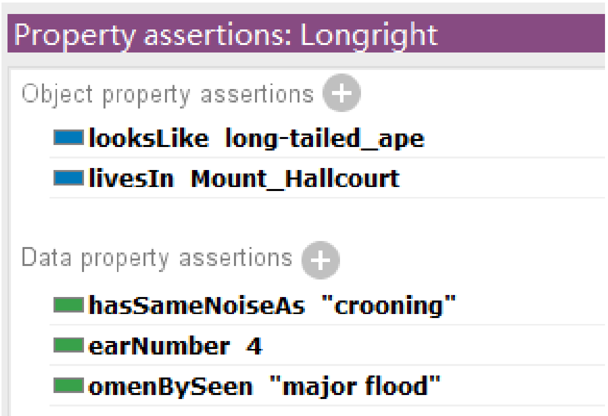


Figure 4.7: The Data Mapping Example of Monster `Longright` in Protege

```
chm-r:Longright    a                    owl:NamedIndividual ,
                                        chm-o:monster ;
                   chm-o:livesIn        chm-r:Mount_Hallcourt ;
                   chm-o:looksLike      dbo:long-tailed_ape ;
                   chm-o:earNumber      4 ;
                   chm-o:hasSameNoiseAs "crooning" ;
                   chm-o:omenBySeen     "major flood" .
```

Listing 14: The Data Mapping Example of Monster `Longright` in Turtle

[2]This monster is extracted from the chapter *Classic of Mountains: Southern*

### 4.5.2 Linking

Interlinking is a task that attaches the same instances from different datasets together by comparing instances [Fan, 2014]. It requires to determine the qualified resource types in our dataset, and then use some specific properties to point to the potentially appropriate matching resources in the external dataset. In this thesis, we select the `Silk-Link Discovery Framework`[Volz et al., 2009] to link our data automatically, which is an open source framework for integrating heterogeneous data sources. We also create some links manually, which is a time-consuming but required task for those inconsistent data.
This process mainly focuses on linking our resources with `DBpedia`, `Wikidata` and `Schema`[3] datasets by using the predicate `owl:sameAs`. `owl:sameAs` is an OWL statement applied for mapping two different URIs which express the same extensional meaning or refer to the same thing. Most animals, colours and few monsters involved in our dataset identify the same individuals as in `DBpedia` and `wikidata`, which indicates that links could be created between individuals in these databases. One example of the instance-level interlinking is the animal `chicken`, which in our dataset is adopted as an instance of the class `animal` to depict the appearance of the monster. The instance `chicken` in our ontology is exactly same to the representation in `DBpedia` and `Wikidata`, hence the property `owl:sameAs` is adopted. In the following statements (see List.15) we could state that the specific URI reference of our dataset refers to the same animal `chicken` as the related URIs in the other two ontologies.

```
<http://chinesemonster/resource/chicken>
   <owl:sameAs>
<http://dbpedia.org/resource/Chicken>
   <owl:sameAs>
<https://www.wikidata.org/wiki/Q780>
```

Listing 15: Instance-level Interlinking Example of Animal `Chicken`

Besides the instance-level interlinking which pertained by the predicate `owl:sameAs`, we also accept the predicates `owl:equivalentClass` and `owl:equivalentProperty` for representing links between classes and properties respectively. For example, we map the class `mountain` in our dataset, which represents the mountains that monsters live in, to the equivalent classes identified by external datasets such as `Schema` and `place`[4] via the predicate `owl:equivalentClass`. The alignment of this example is shown in List.16.

[3] It is available at http://schema.org/
[4] http://lov.okfn.org/dataset/lov/vocabs/place

```
<http://chinesemonster/ontology/mountain>
   <owl:equivalentClass>
<http://schema.org/Mountain>
   <owl:equivalentClass>
<http://purl.org/ontology/places#Mountain>
   <owl:equivalentClass>
<http://dbpedia.org/ontology/Mountain>
```

Listing 16: Class-level Interlinking Example of Class `Mountain`

Results of our dataset linking to the external datasets are performed in Table.4.3, where prefixes of the particular target vocabularies, the number of links we generated between our LD and the target ones as well as the predicates we used to link them are displayed.

| Target Dataset | Predicate | Link Count |
|---|---|---|
| dbpedia | owl:sameAs | 49 |
| wikidata | owl:sameAs | 35 |
| dbpedia | owl:equivalentClass | 4 |
| schema | owl:equivalentClass | 10 |
| place | owl:equivalentClass | 3 |
| wlo | owl:equivalentProperty | 3 |

Table 4.3: Results of Interlinking to Other Ontologies

By generating these relations between our dataset and the external open databases, we interlink our machine-readable data with other existing information sources, thus improving the visibility and usability of our data and facilitating data integration at the web-scale.

## 4.6 Publication

Publishing LD on the Web and making it accessible under an open license is a foundational requirement of LOD practices and is key to implementing the vision of the Web of LOD [Bizer et al., 2008]. The goal of the LD publication process is to make available and discoverable on the Web the generated LD set and its associated ontology [Radulovic et al., 2015].
In this thesis, we have published the metadata of the ontology itself in RDF, which excludes the resources. For the metadata, we consider URIs as identifiers of the ontology and prefixes as abbreviations of the namespaces. Documentations are uploaded as well and will be renewed with the updating of the ontology. In the documenta-

tion, we employ the vocabulary `rdfs:label` to represent the human-understandable identification of each entity, while `rdfs:comment` is utilised to explain the meaning of each term and the role it plays in our ontology. Besides, the mapped data corresponding to each term is published as well to provide a way to apply those vocabularies, which is essential for quickly understanding and reusing the ontology.
Since in this thesis, we only complete the monster part of *Shanhaijing*, and other work will be finished in the future, we choose to publish the ontology of the monster along with other sources on the GitHub[5].

## 4.7 Exploitation

After having published the linked data, we could access it through different systems, including SW browsers and SW search engines [Thorsen and Pattuelli]. The most common method for querying LD is using SPARQL, which is a query language for the RDF. The SPARQL query is composed of triple patterns of variables. The solution consists of binding these variables to entities which are related to each other in the RDF model according to the query structure [Reddy and Kumar].
List.17 is a SPARQL query example that accesses our linked data via SPARQL. In this query, we search for those monsters whose eye colours have been described in *Shanhaijing* and print them out along with their eye colours. We run the query according to Apache Jena, result of which are shown in Table.4.4.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX chm-o: <http://chinesemonster.org/ontology/>

SELECT ?monster ?eyeColor
WHERE{
     ?monster rdf:Type chm-o:monster .
     ?monster chm-o:eyeColor ?eyeColor .
     }
```

Listing 17: SPARQL Query Example of Monster's Eye Colors

[5] *Shanhaijing* ontologies available at https://github.com/aaasteria/chinesemonster/blob/master/ontology.owl and https://github.com/aaasteria/chinesemonster/blob/master/data.ttl

| **Monster** | **Eyecolor** |
|---|---|
| <http://chinesemonster.org/resource/BigYellowsparrwo> | <http://chinesemonster.org/resource/Black> |
| <http://chinesemonster.org/resource/Collar-spoon> | <http://chinesemonster.org/resource/Scarlet> |
| <http://chinesemonster.org/resource/TheGreenBird> | <http://chinesemonster.org/resource/Black> |
| <http://chinesemonster.org/resource/Choir-blend> | <http://chinesemonster.org/resource/Scarlet> |
| <http://chinesemonster.org/resource/YoungYellowsparrow> | <http://chinesemonster.org/resource/Black> |
| <http://chinesemonster.org/resource/Cur-reach> | <http://chinesemonster.org/resource/Scarlet> |
| <http://chinesemonster.org/resource/Cormorant> | <http://chinesemonster.org/resource/Crimson> |
| <http://chinesemonster.org/resource/Green-plough> | <http://chinesemonster.org/resource/White> |

Table 4.4: Query Results for Monster's Eye Colors

## 4.8 Conclusion

In this chapter, we provided the process of building our ontology in detail. We followed the progressive steps to establish our ontology and manage our data, which contains Specification, Modeling, Generation, Publication and Exploration. We have published our metadata of *Shanhaijing* ontology and LD on GitHub.

Chapter 5

# *Shanhaijing* LD Visualization

In this chapter, we introduce an interactive data exploration tool built for the *Shanhaijing* ontology, for the use of which, we have published a video on GitHub[1]. Section 5.1 gives a general overview of this tool, while In Section 5.2, the *Shanhaijing* ontology exploration using the tool is described. Section 5.3 talks about the SPARQL query search function of the tool. In Section 5.4, we show the visualisation of *Shanhaijing* stories, and Section 5.5 is the conclusion of this chapter.

## 5.1 General Overview

The LD interactive explorer tool is a domain specific application created for *Shanhaijing*. It provides query and visualisation functions based on the ontology and data resources that we summarised, extracted and generated from *Shanhaijing*. In this system, users can request related information in *Shanhaijing* and the system will support SPARQL queries as well as visualise the query results. This application is separated into three main parts, the ontology and data exploration part, the SPARQL query writing part and the story visualization part, all of which will be described in the following sections. To explain functions of this application in a more detailed and precise way, we also upload a video on GitHub, which helps to demonstrate our entire system and can be treated as a guideline for users to be familiar with the use of this tool in a short period.
The home page of this application is shown in Fig.5.1. As we introduced in Chapter 2, the content of this book is structured with mythic geography, where elements are located and recorded regarding place and direction. Based on this factor, we decide to build our application on a fictional map[2], which not only suits the theme of the mythology but also offers a way to visualize the narratives of *Shanhaijing*.

[1]The video is accessible at https://github.com/aaasteria/chinesemonster
[2]This map is redrawn based on the picture from http://www.zcool.com.cn/work/ZMjM5NjI1Mjg=.html


Figure 5.1: The Homepage of The Interactive Explorer Tool

## 5.2 *Shanhaijing* Ontology and LD Exploration

In our interactive explorer tool, we provide users with some "quick" search functions to explore our data rather than writing SPARQL queries directly, thus allowing them to be familiar with our ontology and data and easing the way to access and reuse our ontology.

In the application, when the user chooses the "`Ontology`" menu button on the top left of the home page, there comes an input box along with two search buttons. Users could enter any entity names or types here, and a quick search will be executed based on those keywords. Fig.5.2 displays the "quick" way to obtain all classes by typing in a word `class`. In this example, we enter an entity type `class`, with which the system returns URIs of all classes in our ontology shown in the large result box.

After having gained the results, for example, we click the URI link of the `Mountain` class in the result set (the eighth row), and a more detailed description about the chosen class will be presented in a new form (see Fig.5.3). In Fig.5.3, details about the class `Mountain` are demonstrated, where we could figure out the information about its type, superclass, and equivalent classes that it interlinks with other ontologies, etc.

Similarly, when we change the keyword to `ObjectProperty`, the system will return URIs of all entities with the type of `owl:ObejctProperty` (see Fig.5.4). Besides, this system also supports the search of a specific entity. For example, in Fig.5.5, instead of inputting entity types, we enter a specific entity `hasChild`, which is an object prop-


Figure 5.2: Quick Search for All the Classes in *Shanhaijing* Ontology


Figure 5.3: Details of Class `Mountain` Shown by Clicking the URI

erty with an inverse relation to the property `childOf`. As shown in Fig.5.5, results of the `hasChild` keyword search represent detailed descriptions of this property, where its label, comment, type, inverse property as well as equivalent properties are presented.


Figure 5.4: Quick Search for all Object Properties

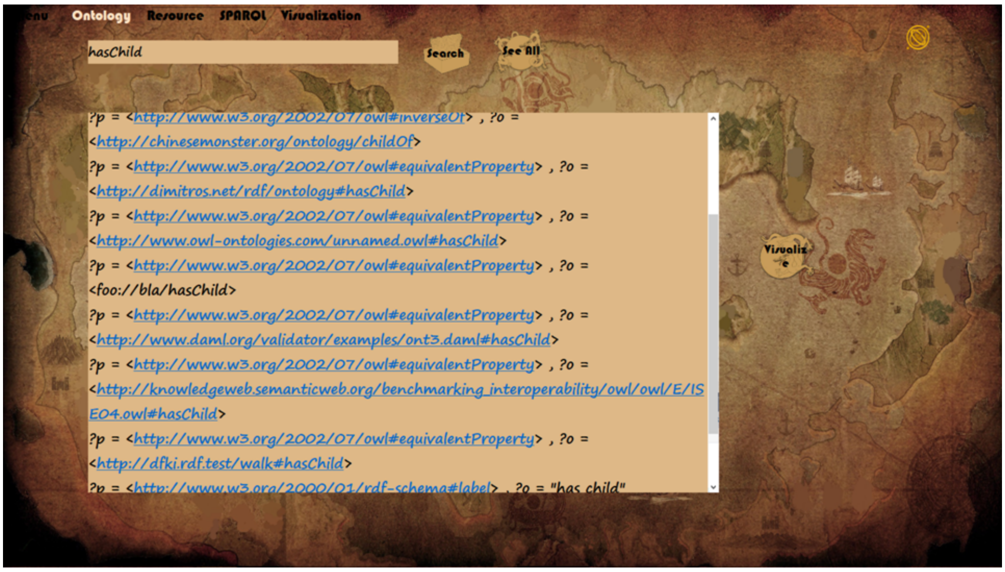


Figure 5.5: Quick Search for the Object Property `hasChild`

Additionally, this system not only supports ontology keyword search but also offers a way to get access to the full version of the ontology. When we click the button `"See All"` on the page, there will be a browser linking to the GitHub site, where we published our ontology. This function is demonstrated in Fig.5.6.

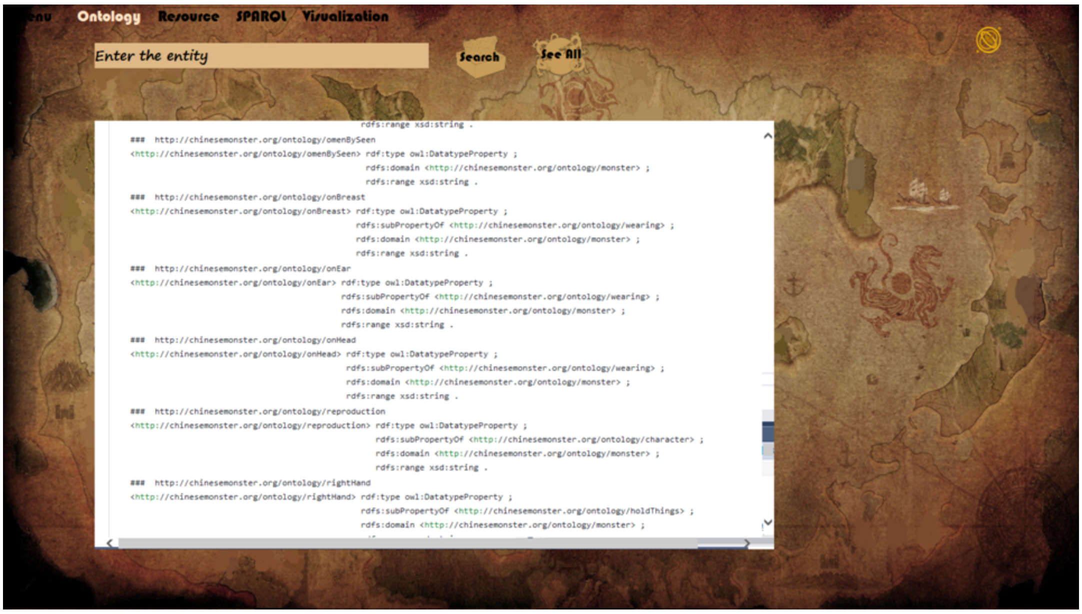


Figure 5.6: Linking to the Published *Shanhaijing* Ontology on GitHub

Regarding the LD exploration functions, they are similar to the functions in the ontology part that we introduced before. Keywords that could be entered for the quick search include both instance types and names. When we decide a specific query instance, for example in Fig.5.7, where we take a monster's name "`ApeHowl(禺虢)`[3]" as an input keyword, the system will return related information of this monster, including its bird-like body, human-like face, and decorations on the ear, etc.
The difference between the ontology query and LD exploration is that when we type in the same keyword, results will be distinct. For instance, we apply `monster` to both the ontology and LD exploration input boxes, the outcome from the ontology search is descriptions of the class `monster`, while in the LD retrieval it turns to the instances of this class. The difference comparisons are displayed in Fig.5.8 and Fig.5.9.
Moreover, the browser function that links to the LD published site is similar to the ontology browser part, which will be omitted to explain here.

[3]This monster is from the chapter *Classic of the Great Wilderness: Eastern*.


Figure 5.7: Quick Search Example for the Monster `ApeHowl`


Figure 5.8: Results of Quick Search Example "monster" in Ontology Exploration Section


Figure 5.9: Results of Quick Search Example "monster" in LD Exploration Section

## 5.3 SPARQL Query Function

In this section, we will introduce the SPARQL query function in our interactive explorer tool. In our system, besides the "quick" search function with keywords, we also allow users to write their SPARQL queries, by which more flexible and variable search could be generated.
To explain this function more specifically, we present a sample SPARQL instance, looking for those monsters and their tails' number if the `tailNumber` feature of a particular monster is mentioned in *Shanhaijing*. List.18 represents the basic SPARQL query that describes the monster and its tail number. The equivalent SPARQL search in our application is illustrated in Fig.5.10. In our system, prefixes are given and displayed at the top field, which is for the convenience of users writing their queries. This system enables users to write queries in the left text box, and when the `Query` button is selected, results will be performed in the right box.
From Fig.5.10 we can see that the returned results are shown by URIs, which are not clear and straightforward enough for users. To address this issue, we employ three charts, the pie chart, the radar chart and the bar chart, by which the query results are depicted more human-readable and obvious. Fig.5.11 illustrates the visual result of the same query with the pie chart, from which we could figure out that among all the monsters whose tail number is mentioned in *Shanhaijing*, those with nine tails account for the largest proportion. Same results visualized by the radar chart and the bar chart are represented in Fig.5.12 and Fig.5.13 respectively. Users are allowed

to switch among different visualization charts to determine the one that suits their preferences.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX chm-o: <http://chinesemonster.org/ontology/>

SELECT ?monster ?tailNumber
WHERE{
     ?monster rdf:Type chm:monster ;
              chm-o:tailNumber ?tailNumber .
     }
```

Listing 18: SPARQL Query Example for Monster's Tail Number

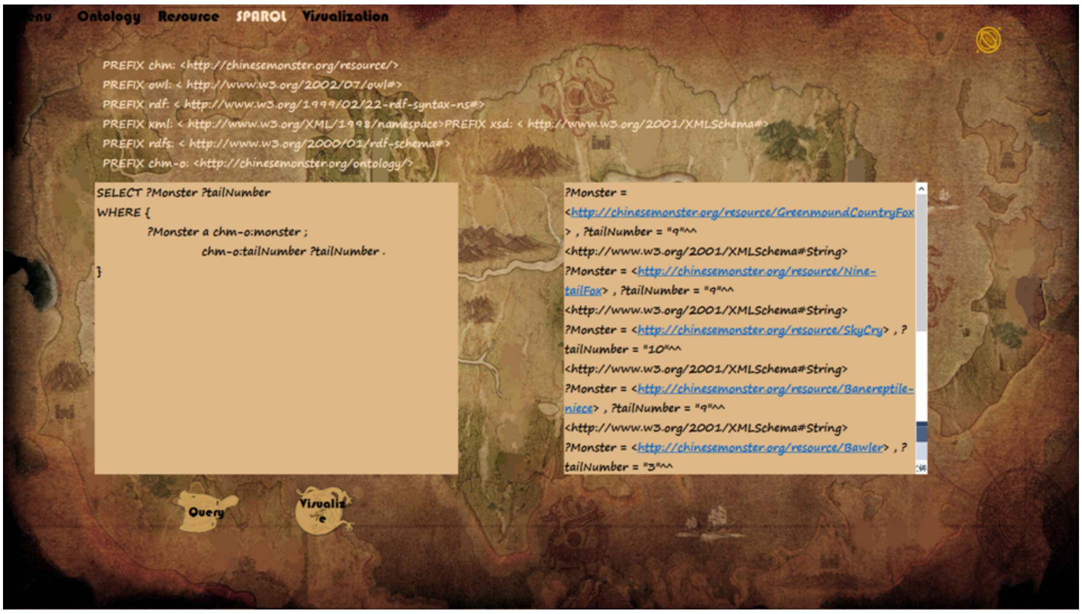


Figure 5.10: SPARQL Query Example for Monster's Tail Number using LD Exploration Tool

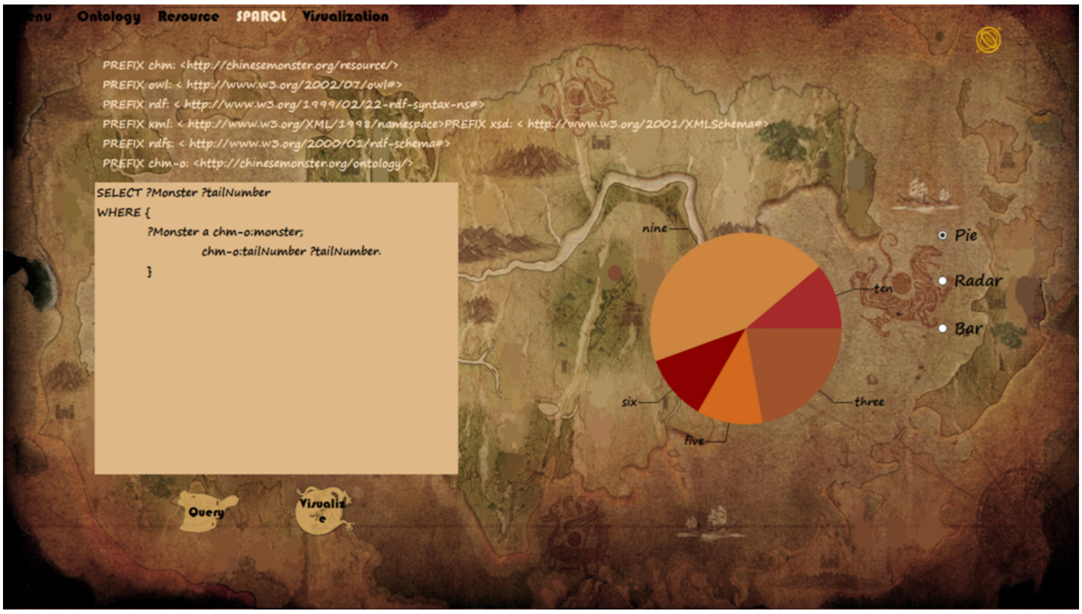


Figure 5.11: Visual Results of Monster's Tail Number with Pie Chart

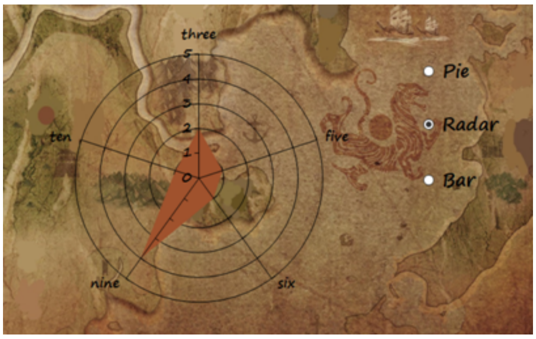


Figure 5.12: Visual Results of Monster's Tail Number with Radar Chart

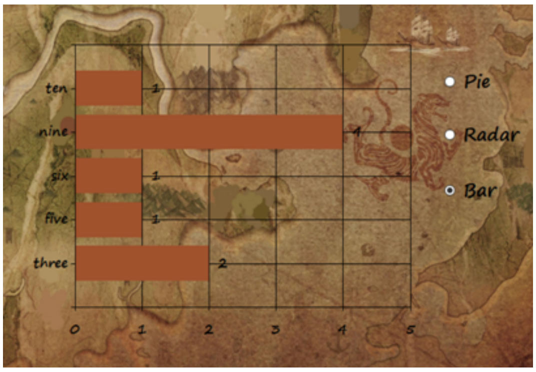


Figure 5.13: Visual Results of Monster's Tail Number with Bar Chart

## 5.4 Data Visualization

To present our data vividly, we provide a graphical version of *Shanhaijing*, where mountains are placed regarding the related direction and location described in this book, and monsters are situated corresponding to the particular mountains where they live. As we mentioned in Chapter 3, *Shanhaijing* is divided into four categories: Classic of the Mountains, Classic of the Seas, Classic of the Great Wilderness, and Classic of Regions Within the Seas. However, the length of the Classic of Mountains is over the sum of the other categories. Therefore, we consider using its five subparts, the Southern Mountains, the Northern Mountains, the Western Mountains, the Eastern Mountains and the Central Mountains along with other classics as our visualisation regions. Hence we have eight separate areas in total making up our visualisation map, each of which could be clicked by users for further exploration. Fig.5.14 is a map of *Shanhaijing* consisting of those eight regions, where users are allowed to choose any parts to learn about, thus enabling them to gain a better understanding about this book as well as our linked data.


Figure 5.14: Visualization Map of Shanhaijing in LD Exploration Tool

To explain the visualisation function more detailed, we take the Southern Mountains part as an example to show the further exploration of *Shanhaijing*. Fig.5.15 is a framing map which contains the geographical information of the Southern Mountains. All the mountains belonging to the Southern Mountains series are listed on this map, following the related direction and location among them. Images of monsters[4]

[4]Images are from http://www.360doc.com/content/13/0217/22/11549880_266235241.shtml and http://www.zcool.com.cn/work/ZMjU4NjQ4MzY=.html

are shown corresponding to the mountains they live in, thus providing a straightforward way for users to gain an initial impression of monsters' appearances and their geographical positions.


Figure 5.15: Map of the Classic of Mountains: Southern

Besides displaying the relative distributions of monsters and mountains, we also encourage users to discover the knowledge of monsters by clicking their images. For instance, when we select the figure of `Nine-tail Fox` on the map which represents the famous monster living in the Mount Greenmound, an identification card comes out involving the information of this monster in *Shanhaijing*. This is illustrated in Fig.5.16. Raw text describing this monster in the book is translated, classified, structured, mapped, interlinked and then transferred into LD. Then we employ SPARQL to query the information of this monster behind the UI, thus showing the results on the message card. From the card, we could know the monster's appearance, English name, Chinese name, habitat, efficacy, etc. We could also find out some exciting features about this monster, such as it looks like a fox but nine tails, and it has a baby's noise but devouring the human.

In our dataset, each instance is allocated with a message card. Instances which link to one resource by objective properties will be shown on the card of this resource, and their names are treated as different colours from other data. In our system, we provide a quick access link to users to move to the linked cards by clicking their names on the previous card. In this way, users could broaden their knowledge about *Shanhaijing* and can have a more clear view of this monster, and our data can be connected as well. In Fig.5.16, the Nine-tail Fox looks like a fox and lives in Mount Greenmound, where `fox` and `Greenmound` are instances of classes in our LD as well.

By selecting the instance's name Mount Greenmound, a card consisting of the information of this mountain is demonstrated (see Fig.5.17). From the card presenting the Mount Greenmound, not only could we gain the knowledge of this mountain, but we are also able to move to other instances which have connections with this mountain. In our system, rather than just connecting data in our dataset, we also link our instances to other resources on the web. The property we applied is `owl:seeAlso`. We still take the Nine-tail Fox in Fig.5.16 as an example. At the bottom of this card is the available resource that connects with our instance. Users are allowed to click the URL of the resource to open the web page in our system. In this example, the resource we linked to is the Nine-tailed Fox in Wikidata (see Fig.5.18), from which users could make more observation about this monster. This function helps us to link our data with the web sources, which is LD rules request. Besides, by this means, users will not be limited to our data, but have the chance to know more about instances in our LD via other resources and could gain a better understanding about *Shanhaijing*.


Figure 5.16: Nine-tail Fox Information Board


Figure 5.17: Mount Greenmound Information Board

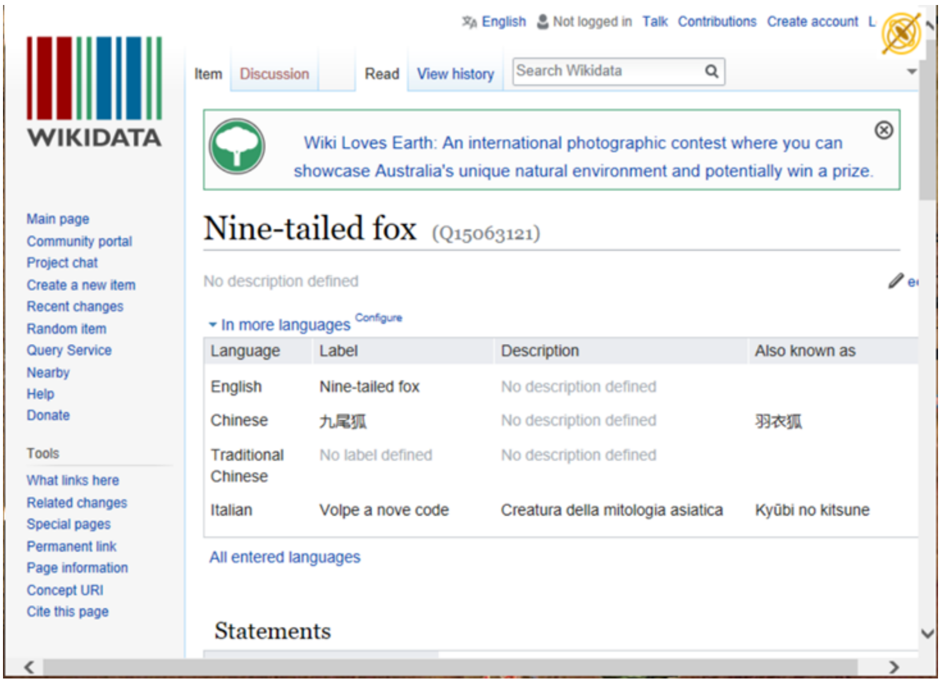


Figure 5.18: Nine-tail Fox Linking Resource

## 5.5 Conclusion

In this chapter, we described an interactive data exploration tool which we created for this thesis. We introduced this tool in several aspects, including the ontology exploration function, which is adopted for users to get familiar with our ontology; the SPARQL query function, which is built for experts and SPARQL users to do further researches based on our ontology; and the visualization function, which is used for demonstrating stories of *Shanhaijing*. Also, the video which shows all the functions of the interactive explorer has been published on GitHub.

Chapter 6

# Case Study

In this chapter, we discuss three use cases from different aspects of *Shanhaijing* to analyze our linked data using SPARQL. Section 6.1 presents the medicinal value of monsters, while Section 6.2 describes the roles of the monsters in divination. In Section 6.3, We analyze some of the potential links between different characteristics of the monster. We conclude this chapter in Section 6.4.

## 6.1 Use Case 1: Medicinal Value of Monsters

Rather than just a mythology book, *Shanhaijing* is an encyclopedia in the Pre-Qin literature. It accounts for a wealth of content, which records plants, animals, minerals, medicines, astronomy, geography, history, religion, ethnic, tribe and so on [Zhao, 2014]. The knowledge of medical history and medication in this book is essential for the study of the germination and evolution of Chinese medicine.

There are more than 120 kinds of drugs recorded in *Shanhaijing*, which cover plants, monsters, and minerals, and which can treat dozens of diseases. This book also consists of a variety of treatment methods, including diet, bathing, wearing, daub, and others. Since in this thesis we only focus on the monster part of *Shanhaijing*, we allow users to search for the medicinal value of monsters with different treatment methods based on our LD. In our data, the efficacy of monsters recorded without any treatment methods in *Shanhaijing* is connected with the specific monsters via the property `chm-o:efficacy`, while others are interlinked to monsters with their particular preparation methods as properties, such as `chm-o:byEaten` and `chm-o:byDosed`, which are the sub-properties of `chm-o:efficacy`. Users are encouraged to apply the SPARQL query in List.19 to query monsters' efficacy in our interactive explorer tool. This query is used for finding out all the monsters which are of medicinal value, no matter what the treatment method is. Table.6.1 shows part of the results of the search.

By this way, users and researchers who are expertizing in this area could access the data that they are looking for quickly and efficiently, and further analyses could be applied based on our LD, thus providing convenience for their research in the future.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX chm-o: <http://chinesemonster.org/ontology/>

SELECT ?monster ?efficacy
WHERE{ {?monster rdf:Type chm-o:monster .
       ?monster chm-o:efficacy ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byEaten ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byDosed ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byReared ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byWorn ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byRiden ?efficacy .}
       }
```

Listing 19: Case Study of Monsters' Efficacy

| **Monster** | **Efficacy** |
|---|---|
| <http://chinesemonster.org/resource/Steal-lard> | against fire |
| <http://chinesemonster.org/resource/ugly-coarse> | against fire |
| <http://chinesemonster.org/resource/Wild-carob> | against misfortune |
| <http://chinesemonster.org/resource/Perch-fish> | cure tumours |
| <http://chinesemonster.org/resource/Three-footedTurtles> | cure swelling |
| <http://chinesemonster.org/resource/Ear-rat> | against hundred poisons |
| <http://chinesemonster.org/resource/Flying-fish> | against weapons |
| <http://chinesemonster.org/resource/Buffalo-reptile> | against fire |

Table 6.1: Results for Monster Efficacy

Instead of searching for the efficacy of all monsters, we could also write simple queries referring to List.19 to focus on how many monsters could be used for a specific illness, or to look for a particular treatment method to figure out the corresponding monsters that could cure diseases in this way. List.20 is a query based on List.19, aiming to calculate the number of monsters that have the same efficacy, which may be valuable for experts who are interested in the medical history and medication of ancient China from 2000 years ago. Visual results of this query are displayed in Fig.6.1, from which we can know that many monsters can be used for preventing malign forces or epidemic diseases, and there even exist monsters useful for contraception, which is the earliest record of contraceptives in China.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX chm-o: <http://chinesemonster.org/ontology/>

SELECT (COUNT(?efficacy) AS ?count) ?efficacy
WHERE{ {?monster rdf:Type chm-o:monster .
       ?monster chm-o:efficacy ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byEaten ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byDosed ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byReared ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byWorn ?efficacy .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:byRiden ?efficacy .}
       } GROUP BY ?efficacy
         ORDER BY ?efficacy
```

Listing 20: Calculation of Monsters' Efficacy


Figure 6.1: Visualization of Monsters' Efficacy Calculation Result

## 6.2 Use Case 2: Monsters in Divination

In the previous section, we have introduced using SPARQL to demonstrate the medical value of monsters, while in *Shanhaijing*, the content of divination is also involved. In this book, many objects especially monsters are taken to foretell good or ill luck, or to discover the sign of a disaster, which is thought as related to the hunting production of ancient people and is also in line with people's superstitious psychology. The divination part is a vital aspect of the study of this book and is a valuable material for researching the culture of that era as well. Due to this reason, we consider this topic as our use case in this section.
There are two types of divination in *Shanhaijing*. The first type is the good omen, which in the book is described as "when the monster appears, there will be a bumper harvest," or "when it comes out, the world will be order and peace." In this thesis, we employ the data property `chm:goodOmen` to link this kind of data to monsters. The other type is the ill omen, which is illustrated in the book with the keywords of the flood, war, drought or panic, etc. Contrast to the good omen, we use `chm:illOmen` to represent this case. Both of the two properties are recorded in our ontology, along with a super-property `chm:omen` as the abstraction.
Users could use the following query (see List.21) to explore the divination omens of monsters. We link monsters with related omens, and summarize them together, thus providing a convenient way to starters who are not much familiar with *Shanhaijing* to have a quick view of this aspect, and offering experts an efficient tool to do further analyses on monsters and their omens as well as explore the potential connections between the monsters' divination signs and the natural phenomena.
Table.6.2 displays part of the results of the query containing both kinds of omens, from which we could know the monsters along with the omens when they appear.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX chm-o: <http://chinesemonster.org/ontology/>

SELECT ?monster ?goodOmen ?illOmen
WHERE{ {?monster rdf:Type chm-o:monster .
       ?monster chm-o:goodOmen ?goodOmen .}
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:illOmen ?illOmen .}
       }
```

Listing 21: Query for Omens of Monsters

| Monster | Good Omen | Ill Omen |
|---|---|---|
| <http://chinesemonster.org/resource/Trickster> | bumper harvest | Null |
| <http://chinesemonster.org/resource/Wonderbird> | order and peace | Null |
| <http://chinesemonster.org/resource/Joinflaw> | Null | major flood |
| <http://chinesemonster.org/resource/Longright> | Null | major flood |
| <http://chinesemonster.org/resource/Duck-wait> | Null | major war |
| <http://chinesemonster.org/resource/LargeSnake> | Null | major drought |
| <http://chinesemonster.org/resource/Far-far> | Null | fraudulent strangers |
| <http://chinesemonster.org/resource/TheMess> | Null | great plague |

Table 6.2: Results for Omens of Monsters

Other search based on the given query can be generated as well. For example, we could calculate the statistic of each omen, which may be worthy for some researchers. This query is realized in List.22 and Table.6.3 presents the results. It is evident from the table that most monsters are considered as taking ill omens, at the most time their appearance has some connections with floods, droughts or wars. This may due to the reason that ancient people have been aware that many animals are more sensitive to perceive natural changes and can effectively forecast disasters, or may because of the lack of a comprehensive understanding of the nature resulting in people's superstitious psychology.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX chm-o: <http://chinesemonster.org/ontology/>

SELECT ?illOmen (COUNT(?illOmen)AS ?illOmenCount) ?goodOmen
 ↪  (COUNT(?goodOmen)AS ?goodOmenCount)
WHERE{ {?monster rdf:Type chm-o:monster .
       ?monster chm-o:illOmen ?illOmen.
       }
UNION {?monster rdf:Type chm-o:monster .
       ?monster chm-o:goodOmen ?goodOmen.}
} GROUP BY ?illOmen ?goodOmen
```

Listing 22: Statistics for Omens of Monsters

| Good Omen | Count | Ill Omen | Count |
|---|---|---|---|
| bumper harvest | 3 | major earthworks | 2 |
| order and peace | 3 | fire | 2 |
| | | panic | 3 |
| | | great plague | 4 |
| | | locusts damage | 1 |
| | | major flood | 8 |
| | | typhoon | 2 |
| | | perish | 1 |
| | | officers being banished | 1 |
| | | major drought | 11 |
| | | water, rain and wind | 1 |
| | | major war | 8 |
| | | fraudulent strangers | 1 |

Table 6.3: Statistical Results for Omens of Monsters

## 6.3 Use Case 3: Connections among Monsters' Characteristics

In this section, we provide two examples to find out the potential connections among monsters' characteristics and make explanations as well as analyses on the phenomenon. Section 6.3.1 introduces the relationship between monsters' dietary characters and their noises, while in Section 6.3.2 we do a research on the relations of monsters' decorations.

### 6.3.1 Connections between Noises and Dietary Characters

During the deep reading, we noticed that many monsters devour humans. The sounds of them are not necessarily similar to a human baby, but most of the monsters whose noise hears like a baby are human-eaters. To prove our guess, we write a SPARQL query (see List.23) to search for our LD. Results that correspond to this query are displayed in Table.6.4.
From the result table, we could figure out that among the monsters whose noise are like a baby's voice, except the `DwarfSturgeon(䱻鱼)` and the `Human-Fish(人鱼)` whose dietary characters are not recorded in *Shanhaijing*, all the others feed on the human. We tried to make a preliminary analysis of this phenomenon, and the reason might because that these monsters imitate the cries of human babies to disguise themselves and attract people's attention. They deceive people to come close and then attack suddenly.
We provide some use cases to show the usage of our LD. However, instead of explaining the phenomenon by ourselves, our LD also has the advantage of facilitating scholars who are expertizing in this area for their further research.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX chm-o: <http://chinesemonster.org/ontology/>

SELECT ?monster ?noise ?dietary
WHERE{ ?monster rdf:Type chm-o:monster .
       ?monster chm-o:noiseLike ?noise.
       FILTER regex(?noise, "baby", "i" )
       OPTIONAL{
       ?monster chm-o:dietary ?dietary.
       }
} ORDER BY ?monster
```

Listing 23: SPARQL Query for Connections between Monsters' Dietary Characters and Noises

| Monster | Noise | Dietary Characters |
|---|---|---|
| <http://chinesemonster.org/resource/Banereptile-niece> | baby | human |
| <http://chinesemonster.org/resource/DwarfSturgeon> | baby | |
| <http://chinesemonster.org/resource/Goat-owl> | baby | human |
| <http://chinesemonster.org/resource/Horse-belly> | baby | human |
| <http://chinesemonster.org/resource/Human-fish> | baby | |
| <http://chinesemonster.org/resource/Joinflaw> | baby | human,reptiles,snakes |
| <http://chinesemonster.org/resource/Malignforce-buzzard> | baby | human |
| <http://chinesemonster.org/resource/Nine-tailFox> | baby | human |
| <http://chinesemonster.org/resource/Notchflaw> | baby | human |
| <http://chinesemonster.org/resource/Rhinoceros> | baby | human |

Table 6.4: Query Results of Connections between Monsters' Dietary Characters and Noises

### 6.3.2 Relations of Monsters' Decorations

In this use case, we provide a query to search for the decorations on the monsters. In our data, decorations are linked to the monster with a series of properties which describes the parts that the decorations are on the monsters. These properties include `chm-o:onEar`, `chm-o:onHead`, `chm-o:onBreast`, `chm-o:underfoot` and `chm-o:inHand`, along with a super-property `chm-o:wearing` as the abstraction. The SPARQL query is available at List.24, while the results are presented in Table.6.5. By the limitation of space, we employ the prefix `chm` to abbreviate the namespace <http://chinesemonster.org/resource/>.
One interesting fact in Table.6.5 is that no matter what colours they are, most decorations on the monsters are snakes, while other objects are quite limited. This phenomenon might not be considered as a coincidence. Due to our analyses, there are

two main reasons to explain this fact. The first reason is that in ancient China, there was a cult of snakes. Snakes may die, but they will not topple over, which is considered as a symbol of strong vitality. The other reason is the worship of fecundity. Ancient people regard snakes as reproductive organs, seeing snakes mating means prosperous. Due to these reasons, many tribes consider snakes as their totem, thus deifying them as spirits and adding elements of snakes to the other powers including the monsters that they do not know much. Therefore, we could figure out the fact in our result set that most decorations on monsters are snakes.

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX chm-o: <http://chinesemonster.org/ontology/>

SELECT DISTINCT ?monster ?onEar ?onHead ?onBreast ?underfoot
 ↪ ?inHand
WHERE{
     {?monster chm-o:onEar ?onEar.}
UNION
     {?monster chm-o:onHead ?onHead.}
UNION
     {?monster chm-o:onBreast ?onBreast.}
UNION
     {?monster chm-o:underfoot ?underfoot.}
UNION
     {?monster chm-o:inHand ?inHand.}
} ORDER BY ?monster
```

Listing 24: SPARQL Query of Decorations on Monsters

| Monster | On Ear | On Head | On Breast | Underfoot | In Hand |
|---|---|---|---|---|---|
| chm-r:ApeHowl | two yellow snakes | | | two yellow snakes | |
| chm-r:ApeStrong | two green snakes | | | two scarlet snakes | |
| chm-r:BlackPeople | | | | | snakes |
| chm-r:BoastFather | two yellow snakes | | | | two yellow snakes |
| chm-r:LordBigWalk | | | | | dagger-axe |
| chm-r:NotcourtHalberdmy | two green snakes | | | two scarlet snakes | |
| chm-r:SlitJet | two green snakes | | | two scarlet snakes | |
| chm-r:SnakeshamanPeople | | | | | club |
| chm-r:StrongGood | | | | | snakes |
| chm-r:TheCorpseOfLavishLiken | two green snakes | | | | |
| chm-r:TheCountryOfGentlemen | | | | | sword |
| chm-r:TheDivineWindBird | | shields and snakes | scarlet snake | snakes | |
| chm-r:TheWonderbird | | shields and snakes | scarlet snake | snakes | |

Table 6.5: Query Results of Decorations on Monsters

## 6.4 Conclusion

In this chapter, we provided three use cases to query different aspects of our ontology. Case 1 looked for the medicinal value of monsters, which is an essential aspect of *Shanhaijing*. We introduced Case 2 in Section 6.2, which discussed monsters' roles in divination. We also analyzed possible reasons for this phenomenon. In section 6.3, we focused on finding out potential links between monsters' characteristics, and made analyses based on our results. SPARQL queries of these cases have been published on GitHub.

Chapter 7

# Conclusion

In this chapter, we conclude this thesis, evaluate our contributions and discuss the future work. Section 7.1 represents the main contributions that we have made in this thesis. In Section 7.2, we evaluate our work and describe the limitations of this thesis. While future work discussion is presented in Section 7.3.

## 7.1 Summary of Contributions

In this section, we revise the main contributions of this thesis, which contains building a structured dataset of *Shanhaijing*, producing an innovate ontology based on the dataset, generating linked data of the book, and creating an interactive explorer tool for the linked data.

### 7.1.1 Structured Dataset Production

In this thesis, because of the lack of an available dataset which consists of the structured data of *Shanhaijing* that could be employed for our research, we decided to create one by ourselves. We generated valuable terms based on the raw texts of descriptions of monsters in the book during the deep reading procedure, then summarised and manipulated those terms to form a structured dataset. Our dataset covers monsters in *Shanhaijing*, along with their characteristics, habitats, expressions of their appearances, and properties that link these entities together. This dataset is innovative and might be treated as valuable for other research of *Shanhaijing* as well.

### 7.1.2 Ontology Building

The number of mythological ontologies is not as rich as expected, and most of them are created for Western folk tales. Due to the differences between the narratives of Eastern and Western myths, reusing existing ontologies might not be an efficient choice. Hence we built our ontology from existing ontologies in the same domain, along with some ontologies from other domains which have similar properties or classes to our entities. We extracted entities from both the traditional Chinese text and the English translation and determined which terms to be adopted by our own.

We also mapped our entities to several published ontologies, thus interlinking our ontologies to them. Our ontology metadata has been published on the GitHub.

### 7.1.3 Linked Data Generation

We mapped the structured data set to the ontology, by which we generated the LD of the monster aspect of *Shanhaijing*, which can be queried by SPARQL. We linked our data to various Linked Open Data, and have published it on GitHub. Our LD follows the four principles for publishing LD and can be considered as five-star LD evaluated by Tim Berners-Lee's five-star ranking system.

### 7.1.4 The Interactive Explorer Tool

To demonstrate our LD and introduce *Shanhaijing* to readers, we built an interactive data explorer. This tool allows users who are not familiar with our LD or SPARQL to do a quick search by inputting singles words, and experts to write complicated queries by themselves as well as visualise their results. We also created a visualisation function, which provides a graphical version of our LD and *Shanhaijing*. Besides, users can look through ontologies and resources that our LD linked to via this tool as well. A video of displaying functions of the explorer has been published on GitHub, along with some SPARQL queries which can be reproduced by users.

## 7.2 Evaluations

This thesis is based on the Chinese mythology book *Shanhaijing*, similar work of this book is almost non-existent in this area. In this thesis, we chose the monster aspect of *Shanhaijing* as our research domain, for which we have created a structured dataset, a specific ontology, five-star LD, and an interactive data explorer. We have also published them on the GitHub. Our published work will allow future generations of scholars to benefit not only from our analyses and findings but also to be able to verify them through rerunning the workflow on the same dataset. Future researchers will also be in a position to benefit from, reuse, and redevelop our processes, tools, and data for their research.

There are also some limitations which need to be improved in the future. This thesis focused on the monster aspect of *Shanhaijing*, which did not cover all the content of this book, and is far away from the whole Chinese mythology. Due to the limited period of this thesis, some wonderful parts such as deities and mythic plants of this composition are not involved, but they also occupy essential positions and need to be contained in the future. In this thesis, because of various kinds of instances of some of the monster's characteristics, such as `noiseLike:baby` and `noiseLike:wood_being_split`, we adopted strings to represent values of these characteristics rather than entities, which is not conducive to reuse or interlink to other ontologies. A better solution needs to be carried out in the future to address this issue. Regarding the ontology interlinking, we linked our LD with some well-known or domain-specific ontologies, rather than all of the ontologies that we could find, which is not enough. Additionally, our interactive data explorer is built on the dot-Net Framework, which is not as convenient and accessible for interacting as a web page. Because of time constraints and lack of relevant skills, we did not change our explorer, and this will be considered in the future.

## 7.3 Future Work

In the future, based on our current work, we will use LD methodologies to complete the analyses of all the contents of *Shanhaijing*. New ontologies will be generated from the ontology built in this thesis, and structures will be redetermined and improved to solve those limitations. More ontologies that could be reused or interlinked to will be considered to increase the number of linked elements of our ontology. We will establish a website to store and demonstrate our LD and ontology, the visualisation of contents of *Shanhaijing* will also be contained on the website. In this way, we will provide scholars with a complete tool, reusable ontologies and LD of *Shanhaijing* for their analyses and research.

## 7.4 Conclusion

In this thesis, we use LD methods to do the textual analyses and visualization of a Chinese mythology book, *Shanhaijing*. In Chapter 1, we introduced this thesis briefly. Chapter 2 consists of descriptions of *Shanhaijing*, which is the object we focuses on in our research. Chapter 3 represents preliminaries of the techniques we adopted in this thesis, the background of these methods are also contained. In Chapter 4, we discussed the process of building our ontology for *Shanhaijing*, where the procedure of generating LD is involved as well. Chapter 5 provides a brief overview of the interactive data explorer we created for visualize our LD and contents of this book, while in Chapter 6 three use cases from our LD are mentioned. Finally, in Chapter 7, we evaluated and concluded our work.

# Bibliography


Aasman, J., 2006. Allegro Graph: RDF Triple Database. *Cidade: Oakland Franz Incorporated*, 17 (2006). (cited on page 23)

Abele, A.; McCrae, J. P.; Buitelaar, P.; Jentzsch, A.; and Cyganiak, R., 2017. Linking Open Data Cloud Diagram 2017. http://lod-cloud.net/. (cited on pages xiii, 24, and 25)

Ahmed, Z. and Gerhard, D., 2010. Role of Ontology in Semantic Web Development. *arXiv preprint*, (2010). (cited on page 27)

Akoumianakis, D., 2009. *Virtual Community Practices and Social Interactive Media: Technology Lifecycle and Workflow Analysis: Technology Lifecycle and Workflow Analysis*. IGI Global. (cited on page 12)

Alam, M., 2015. *Interactive Knowledge Discovery over Web of Data.* Ph.D. thesis, Loria & Inria Grand Est. (cited on page 18)

Alhawiti, M. and Abdelhamid, Y., 2016. Collection of Web Multimedia Teaching Resources Using Ontology. *International Journal of Information and Education Technology*, 6, 4 (2016), 269. (cited on page 2)

Almas, B., 2017. Perseids: Experimenting with Infrastructure for Creating and Sharing Research Data in the Digital Humanities. *Data Science Journal*, 16 (2017). (cited on page 1)

Almas, B.; Beaulieu, M.-C.; Buckingham, T.; Baumgardt, F.; Cerrato, L.; and Arundel, J., 2017. Perseids-A Collaborative Editing Platform for Source Documents in Classics. http://sites.tufts.edu/perseids/. (cited on page 1)

Ashraf, J., 2013. *A Semantic Framework for Ontology Usage Analysis*. Ph.D. thesis, Citeseer. (cited on page 17)

Bechhofer, S.; van Harmelen, F.; Hendler, J.; Horrocks, I.; McGuinness, D. L.; Patel-Schneider, P. F.; and Stein, L. A., 2004. OWL Web Ontology Language Reference. Technical report, World Wide Web Consortium. (cited on pages 19 and 31)

Berners-Lee, T., 1998. Cool URIs Don't Change. (1998). (cited on page 20)

Berners-Lee, T., 2002. What Do HTTP URIs Identify? https://www.w3.org/DesignIssues/HTTP-URI.html. (cited on page 1)

Berners-Lee, T., 2006. Linked Data - Design Issues. https://www.w3.org/DesignIssues/LinkedData.html. (cited on page 24)

Berners-Lee, T.; Hendler, J.; and Lassila, O., 2001. The Semantic Web. *Scientific american*, 284, 5 (2001), 34–43. (cited on pages 1, 2, and 11)

Birrell, A., 1999. *The Classic of Mountains and Seas*. Penguin. (cited on pages 3, 5, 13, and 30)

Bizer, C.; Cyganiak, R.; and Heath, T., 2008. How to Publish Linked Data on the Web. http://wifo5-03.informatik.uni-mannheim.de/bizer/pub/LinkedDataTutorial/. (cited on page 40)

Burrows, J., 2002. 'Delta': A Measure of Stylistic Difference and a Guide to Likely Authorship. *Literary and linguistic computing*, 17, 3 (2002), 267–287. (cited on page 2)

C, S. and RV, R., 2017. Developing an Ontology for Greek Mythology. *The Electronic Library*, 36 (2017). (cited on page 3)

Chen, H., 1995. 狐狸精原型的文化阐释. 北方论丛, (1995), 38–43. (cited on page 3)

Clark, K. G.; Feigenbaum, L.; and Torres, E., 2006. SPARQL Protocol for RDF. https://www.w3.org/TR/rdf-sparql-protocol/. (cited on page 22)

Coates, T.; Connolly, D.; and Dack, D., 2001. URIs, URLs, and URNs: Clarifications and Recommendations 1.0. https://www.w3.org/TR/uri-clarification/. (cited on page 12)

Complicated, S. T. R. G., 2009. Linked Data-Connect Distributed Data Across the Web. (2009). (cited on page 23)

DBpedia, 2014. The DBpedia Ontology. http://mappings.dbpedia.org/server/ontology/classes/. (cited on page 29)

Du, G., 1988. 墉城集仙录. 文渊阁四库全书, , 899 (1988), 44. (cited on page 9)

Du, K., 2017. Authorship of Dream of the Red Chamber: A Topic Modeling Approach. *Digital Humanities*, (2017). (cited on page 2)

Dutta, B., 2017. Examining the Interrelatedness between Ontologies and Linked Data. *Library Hi Tech*, 35, 2 (2017), 312–331. (cited on page 27)

Fan, Z., 2014. *Concise Pattern Learning for RDF Data Sets Interlinking*. Ph.D. thesis, Université de Grenoble. (cited on page 39)

Franzini, E.; Franzini, G.; Rotari, G.; Jander, M.; Büchler, M.; Solhdoust, M.; and Pannach, F., 2015. Digital Breadcrumbs of Brothers Grimm. https://www.etrap.eu/digital-breadcrumbs-of-brothers-grimm/. (cited on pages 1 and 2)

Fu, G. and Rao, W., 2015. An Introduction of Knoodl. (cited on page 21)

Gao, W., 2012. *An Approach to Formalizing Ontology Driven Semantic Integration: Concepts, Dimensions and Framework*. The Florida State University. (cited on page 23)

Gomez-Perez, A.; Fernandez-Lopez, M.; and Corcho, O., 2004. Ontological Engineering: With Examples from the Areas of Knowledge Management, E-commerce and Semantic Web. *London: Springer Verlag*, (2004). (cited on page 27)

Gruber, T. R., 1993. A Translation Approach to Portable Ontology Specifications. *Knowledge acquisition*, 5, 2 (1993), 199–220. (cited on page 16)

Guan, H., 1980. 窦娥冤. 艺术中国网. (cited on page 9)

Guo, P., 2017. 国学基本典籍丛刊：宋本山海经. National Library of China Publishing House, 1 edn. ISBN 9787501362523. (cited on page 4)

Güting, R. H., 1994. GraphDB: Modeling and Querying Graphs in Databases. In *VLDB*, vol. 94, 12–15. (cited on page 23)

Haase, P.; Lewen, H.; Studer, R.; Tran, D. T.; Erdmann, M.; dAquin, M.; and Motta, E., 2008. The Neon Ontology Engineering Toolkit. *WWW*, (2008). (cited on page 21)

Hao, Y., 2016. 山海经笺疏. China Zhigong Publishing House, 1 edn. ISBN 9787514509762. (cited on page 5)

Heath, T. and Bizer, C., 2011. *Linked Data: Evolving the Web Into a Global Data Space*. Morgan & Claypool Publishers. (cited on page 14)

Horridge, M.; Knublauch, H.; Rector, A.; Stevens, R.; and Wroe, C., 2004. A Practical Guide to Building OWL Ontologies Using the Protégé-OWL Plugin and CO-ODE Tools Edition 1.0. *University of Manchester*, (2004). (cited on page 22)

Hu, Y., 2003. 纵观海内外《山海经》研究五十年. 福建师大福清分校学报, (2003), 45–52. (cited on page 8)

Ibrahim, A. M.; Hashi, H. A.; and Mohamed, A. A., 2013. Ontology-driven Information Retrieval for Healthcare Information System: A Case Study. *International Journal of Network Security & Its Applications*, 5, 1 (2013), 61. (cited on page 2)

Idenhen, K., 2013. Introducing OpenLink Virtuoso: Universal Data Access without Boundaries. (cited on page 23)

Jena, A., 2007. Semantic Web Framework For Java. (cited on page 23)

Katsumi, M. and Grüninger, M., 2016. What Is Ontology Reuse? In *FOIS*, 9–22. (cited on page 21)

Klyne, G.; Carroll, J. J.; and McBride, B., 2004. Resource Description Framework (RDF): Concepts and Abstract Syntax. https://www.w3.org/TR/rdf-concepts/. (cited on pages 1 and 12)

Lassila, O. and Swick, R. R., 1997. Resource Description Framework (RDF) Model and Syntax. https://www.w3.org/TR/WD-rdf-syntax-971002/. (cited on page 12)

Lau, R. W. H. and Li, F. W. B., 2006. Advances in Web-based Learning. *International Journal of Distance Education Technologies*, 4, 1 (2006), I. (cited on page 2)

Lee, S.-A., 2011. Lures and Horrors of Alterity: Adapting Korean Tales of Fox Spirits. *International Research in Children's Literature*, 4, 2 (2011), 135–150. (cited on page 3)

Li, E. and Chan, Y. K., 2012. *Chinese Literature*. Singapore : Asiapac Books, 2012. ISBN 9812296433 (pbk.). (cited on pages 8 and 9)

Liu, C.-L.; Jin, G.-T.; Wang, H.; Liu, Q.-F.; Cheng, W.-H.; Chiu, W.-Y.; Tsai, R. T.-H.; and Wang, Y.-C., 2015. Textual Analysis for Studying Chinese Historical Documents and Literary Novels. In *Proceedings of the ASE BigData & SocialInformatics 2015*, 30. (cited on page 2)

Lonsdale, D.; Embley, D. W.; Ding, Y.; Xu, L.; and Hepp, M., 2010. Reusing Ontologies and Language Components for Ontology Generation. *Data & Knowledge Engineering*, 69, 4 (2010), 318–330. (cited on page 21)

Mäkelä, E.; Hypén, K.; and Hyvönen, E., 2013. Fiction literature as linked open data-The BookSampo dataset. *Semantic Web*, 4, 3 (2013), 299–306. (cited on page 1)

Mäkelä, E. and Hyvönen, E., 2011. BookSampo - Semantic Portal for Finnish Fiction Literature. https://seco.cs.aalto.fi/applications/kirjasampo/. (cited on page 1)

Maniraj, V. and Sivakumar, R., 2010. Ontology Languages – A Review. *International Journal of Computer Theory and Engineering*, 2 (2010). (cited on page 17)

Michael, T., 2001. Shan Hai Ching; Or, the Classic of Mountains and Seas. (cited on pages 8 and 9)

Modoni, G.; Sacco, M.; and Terkaj, W., 2014. A Survey of RDF Store Solutions. (cited on page 23)

Musen, M. A., 2015. The Protégé Project: A Look Back and a Look Forward. *AI matters*, 1, 4 (2015), 4–12. (cited on page 4)

Neches, R.; Fikes, R. E.; Finin, T.; Gruber, T.; Patil, R.; Senator, T.; and Swartout, W. R., 1991. Enabling Technology for Knowledge Sharing. *AI magazine*, 12, 3 (1991), 36. (cited on page 16)

Noy, N. F.; McGuinness, D. L.; and others, 2001. Ontology Development 101: A Guide to Creating your First Ontology. (cited on page 22)

Nozaki, K., 1961. *Kitsuné: Japan's Fox of Mystery, Romance & Humor*. Hokuseido Press. (cited on page 3)

Nurmikko-Fuller, T., 2014. Assessing the Suitability of Existing OWL Ontologies for the Representation of Narrative Structures in Sumerian Literature. (2014). (cited on page 1)

O'connor, M. J.; Halaschek-Wiener, C.; and Musen, M. A., 2010. Mapping Master: A Flexible Approach for Mapping Spreadsheets to OWL. In *International Semantic Web Conference*, 194–208. (cited on page 38)

Pan, J. Z.; Vetere, G.; Gomez-Perez, J. M.; and Wu, H., 2017. *Exploiting Linked Data and Knowledge Graphs in Large Organisations*. Springer. (cited on pages 21, 28, and 36)

Pattuelli, M. C.; Provo, A.; and Thorsen, H., 2015. Ontology Building for Linked Open Data: A Pragmatic Perspective. *Journal of Library Metadata*, 15, 3-4 (2015), 265–294. doi:10.1080/19386389.2015.1099979. https://doi.org/10.1080/19386389.2015.1099979. (cited on pages 27 and 36)

Peinado, F.; Gervás, P.; and D\'\iaz-Agudo, B., 2004. A Description Llogic Ontology for Fairy Tale Generation. In *Procs. of the Workshop on Language Resources for Linguistic Creativity, LREC*, vol. 4, 56–61. (cited on pages 1 and 3)

Protege Wiki, 2018. Protege Ontology Library. https://protegewiki.stanford.edu/wiki/Protege_Ontology_Library. (cited on page 29)

Prud'hommeaux, E. and Seaborne, A. SPARQL Query Language for RDF. https://www.w3.org/TR/rdf-sparql-query/. (cited on page 1)

Pu, S., 1988. *Strange Tales of Liaozhai*. Cheng & Tsui. (cited on page 9)

Qi, G.; Tang, J.; Du Jiefeng; Pan, J. Z.; and Yu, Y., 2013. *Linked Data and Knowledge Graph*. Springer-Verlag. ISBN 9783642540257. doi:10.1007/978-3-642-54025-7. (cited on page 22)

Qiu, A., 2008. New Classic of Mountains and Seas I. (cited on page 2)

Radulovic, F.; Poveda-Villalón, M.; Vila-Suero, D.; Rodr'iguez-Doncel, V.; Garc'ia-Castro, R.; and Gómez-Pérez, A., 2015. Guidelines for Linked Data Generation and Publication: An Example in Building Energy Consumption. *Automation in Construction*, 57 (2015), 178–187. (cited on page 40)

RDF Working Group, 2014. RDF. https://www.w3.org/RDF/. (cited on page 1)

Reddy, B. R. K. and Kumar, P. S. Efficient Approximate SPARQL Querying of Web of Linked Data. (cited on page 41)

Schmachtenberg, M.; Bizer, C.; and Paulheim, H., 2014. Adoption of the Linked Data Best Practices in Different Topical Domains. In *The Semantic Web – ISWC 2014*, 245–260. Springer International Publishing, Cham. (cited on page 23)

Slama, O., 2017. *Flexible Querying of RDF Databases: A Contribution Based on Fuzzy Logic*. Ph.D. thesis, Rennes 1. (cited on page 13)

Song Yu, 1977. 高唐赋. 文选, 264 (1977). (cited on page 9)

Spahiu, B., 2017. Profiling Linked Data. (2017). (cited on page 11)

Syamili, C. and Rekha, R. V., 2018. Developing an Ontology for Greek mythology. *The Electronic Library*, 36, 1 (2018), 119–132. (cited on page 1)

Szász, B.; Fleiner, R.; and Micsik, A., 2016. Linked Data Enrichment with Self-unfolding URIs. In *2016 IEEE 14th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, 305–309. doi:10.1109/SAMI.2016.7423025. (cited on page 20)

Takako, T., 1999. Zusetsu Hyakki Yagyō Emaki o Yomu. (cited on page 2)

Thompson, S., 1989. *Motif-index of Folk-literature: a Classification of Narrative Elements in Folktales, Ballads, Myths, Fables, Mediaeval Romances, Exempla, Fabliaux, Jest-books and Local Legends*, vol. 4. Indiana University Press. (cited on page 2)

Thorsen, H. and Pattuelli, M. C. Ontologies in the Time of Linked Data. *NASKO*, 5, 1 , 42750. (cited on page 41)

Tilahun, B. and Kauppinen, T., 2012. Potential of Linked Open Data in Health Information Representation on the Semantic Web. In *SWAT4LS*. (cited on page 11)

Τζανακη, Β., 2016. *Μλτη Απόδοση Κατανμημνων Βασων Δδομνων Για Γραου Για Διαχιριση Διασυνδδμνων Δδομνων*. (2016). (cited on page 14)

Vasant, a. and Voropai, N., 2016. *Sustaining Power Resources through Energy Optimization and Engineering*. ISBN 9781466697553. doi:10.4018/978-1-4666-9755-3. (cited on page 16)

Villazón-Terrazas, B.; Vilches-Blázquez, L. M.; Corcho, O.; and Gómez-Pérez, A., 2011. Methodological Guidelines for Publishing Government Linked Data. In *Linking government data*, 27–49. Springer. (cited on pages 21 and 27)

Volz, J.; Bizer, C.; Gaedke, M.; and Kobilarov, G., 2009. Silk-A Link Discovery Framework for the Web of Data. *LDOW*, 538 (2009). (cited on pages 4 and 39)

Whetzel, P. L.; Noy, N. F.; Shah, N. H.; Alexander, P. R.; Nyulas, C.; Tudorache, T.; and Musen, M. A., 2011. BioPortal: Enhanced Functionality via New Web Services from the National Center for Biomedical Ontology to Access and Use Ontologies in Software Applications. *Nucleic acids research*, 39, suppl_2 (2011), W541–W545. (cited on page 29)

Wu, R., 1983. 山海经广注. 文渊阁四库全书*(第1042 册)*, (1983). (cited on pages xiii and 8)

Xu, Z., 2009. 封神演义. Zhejiang Publishing United Group Digital Media Co., Ltd. (cited on page 9)

Xueqin, C., 2011. *Dream of the Red Chamber*. Tuttle Publishing. (cited on page 2)

Yarandi, M., 2013. *Semantic Rule-based Approach for Supporting Personalised Adaptive E-Learning*. Ph.D. thesis, University of East London. (cited on page 12)

Yu, L., 2011. *A Developer's Guide to the Semantic Web*. Springer Science & Business Media. (cited on page 11)

Yuan, K., 1980. *Shan Hai Jing Jiao Zhu*. Shang hai gu ji chubanshe. (cited on pages 3, 4, 13, and 30)

Zhang, Y. and Qin, Y., 1997. 白话山海经. 三秦出版社. (cited on page 5)

Zhao, T., 2014. Totem Worship in Shanhaijing Xishanjing. *Open Journal of Historical Studies*, 02 (5 2014), 84–89. (cited on page 57)

Zhuangzi and Palmer, M. J., 1996. *The Book of Chuang Tzu*. Penguin. (cited on page 9)

尾崎勤, 2004. 怪奇卷. と中日用>,《汲古, (2004), 68–69. (cited on page 2)