Master's Thesis

Adaptive Security Awareness Training Using Linked Open Data Datasets

1610412 TAN ZHEYU

Supervisor	Research Associate Professor Razvan Beuran
Main Examiner	Research Associate Professor Razvan Beuran
Examiners	Professor Yasuo Tan
	Associate Professor Shinobu Hasegawa
	Associate Professor Yuto Lim

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)

February 2019

Abstract

Along with the extraordinary development of technologies, people are getting more and more cyberrelated, and their daily life is exposed to kinds of cyberattacks, such as password attacks, malware attacks and so on. Cybersecurity is no longer an issue discussed only between the professionals or technologists, but it is also closely related to ordinary people. Wombat Security Technologies conducted a survey revealed that ransomware is an unknown concept to nearly two-thirds of employees. Almost 95% of cybersecurity attacks are due to human error. As much as 54% of companies have experienced one or more attacks in the last 12 months, and the number rises every month. A considerable amount of money spent on cybersecurity to protect people, company and organization from cyberattacks. Expensive and sophisticated systems won't play much good without the human factotr which is the main vulnerability in cybersecurity. It has great significance to give people cybersecurity awareness training. The innovative idea of this research is to apply concept map for cybersecurity awareness training. For adaptive cybersecurity awareness training, two issues can't be neglected, as follows.

1. Firstly, what concepts should used to teach people in awareness training?
2. Secondly, what specific question should be choosen?
3. Last but not least, how to teach people efficiently?

1. *Concept map construction*

This research proposes a way to construct Computer security concept map from the LOD database DBpedia, dynamically and timely updated, and extracted sub concept map from this big concept map.

DBpedia was constructed by extracting data from Wikipedia which has extensive topic coverage, and it is possible to get much further training materials since it interlinked with other kinds of open datasets. The knowledge in DBpedia is in RDF graph which can be accessed by public SPARQL endpoint. This research uses SPARQL querying data from DBpedia and returns results in JSON format to build the concept map. The query in this research uses an important property: $A\ skos : broader\ B$ which represent hierarchical relationships between two concepts, and B has a more general and broader meaning than A. Broader concepts are typically rendered as parents in a concept hierarchy. The query strategy is important if we aimed at building an efficient and useful concept map for cybersecurity awareness training. In this research, we reach the descendent concepts of Computer security with Breadth-first algorithm in level order. The descendent concepts of Computer security are enormous. This research shows several Computer security maps with various depth.

In the course of building a concept map, along with depth growing, the number of nodes/concepts increase rapidly and the time for accessing the Internet growing immediately, especially, when on the relatively large depth, the time for accessing the Internet is almost 779 times slower than the execution time. Timely update the big concept map make the cybersecurity materials is fresh. Constructed sub concept map from the big one save much time in the practical training.

2. *Relevance estimation and filtering on concept map*

This research proposed ways to process the built concept map. Employ the PageRank algorithm to calculate the importance of each concept node and do the filtering on concept map for the next adaptive training.

This research constructed the hierarchy concept map from the DBpedia Categories. The sub concept map generated from the Computer security map may still be large, for example, the Malware sub concept map contains 54 concept nodes which may not all useful or so important for the training. Even they are all useful ideally; we still need to decide which concepts have higher priority for Malware adaptive training. This research estimates the concept importance/relevance by applying PageRank algorithm on the Computer security map. Google uses the PageRank algorithm to determine the relevance or importance of a page, and the importance of a page is determined by the number of links going out of this page. In this research, a concept map is a

set of interlinked concept nodes. We assume that the importance of a concept node is determined by the number of linked concept nodes going out of this concept node. The concept node has higher PageRank value considered has higher importance/relevance for adaptive training.

Next, we need to employ filtering on concept map since not every concept has a definition in DBpedia Articles and so related to the training keyword. At first, this research directly checks each concept has a definition or not. But in the late of this research, when we do the exact example with Computer worms, we found out that the Computer worms do not have a definition, but the Computer worm does have a definition. Without any doubt, no matter Computer worm or Computer worms are useful for awareness training. This unexpected result led us to improve the filter algorithm by linking concept in Categories with the concept in Articles first. The link from Articles to Categories is represented as $dct : subject$. After linking, take concept Malware in Categories as an example, it has multiple results in Articles. Using a NPL library: fuzzywuzzy to find the most approximate string matching of the keyword. If the keyword has a linking concept in Articles, it has a definition in Articles. There is another case needed to be considered, that is, not every concept is relevant to practical training. For example, concept Digimon is in the concept map built from the keyword Malware ("Digimon" is the grandchildren concept of "Malware"). But it is an instance of a Game class, it is irrelevant to the cybersecurity awareness training. We filter this kind of concept based on DBpedia class and property.

3. *Using concept map conduct adaptive training*

We proposed a simple way to do adaptive training and implemented an adaptive awareness training system prototype. The processed concept map combined with the simple learner model provided the idea of the adaptive training. Question creation and text processing made training system into actual practice.

Using the filtered concepts to generate questions. This research provided multiple-choice questions. Creat questions using the definition queried from Articles. The stem of the question is in a straight form of $What\ is\ concept$?, where the concept is the given keyword from the learner. The correct answer to the question is coming from the concept itself. The incorrect answers to the question are coming to form the concepts that on the same level of the keyword concept on the Computer security tree. Both choices are in a straight form of "$...is\ concept\ definition$.", replace the concept in concept definition with "..." by using python regular expression operations to do the text processing.

The processed concept map combined with the simple learner model pro-

vided the idea of the adaptive training. The learner model present in this research using a straightforward version to interact with the learner and adaptive awareness learning system, which reveals the understandings and misunderstanding to correct knowledge. In pratical training, the system prototype in this research set a threshold of the number of the training concepts to six, this threshold can be modified easily.

Update algorithm bring concept map and learner model together to determine which training content should be given next. In this research, adaptive awareness training contains several small quizzes. Each quiz consists of four questions — the count of quizzes determined by the size of the sub concept map and the knowledge of the learner. In initialize of the first quiz, the update algorithm traverse the sub concept map level by level. In each level, the concept with higher PageRank is the priority to be selected. And after each quiz, update the learner model based on the feedback results from the learner. Then, update algorithm select training concepts based on the learner model and the sub concept map again, the correctly answered concept should not appear in the next quiz, until the learner answered all questions right.

Tradition adaptive learning system usually divided into Expert model, Learner model, Instructive model, and Instructional environment. The adaptive system prototype in this research implemented all the primary function of the four modules. For the expert model, the prototype can dynamically and timely update the training materials and extend the training materials. For the learner model, this prototype employs a straightforward one to track the learner. For instructive model, this prototype combines the processed concept and feedbacks from learner to provide the next question. As the survey did in research background, it is innovative to conduct adaptive training in this way. For the instructional environment, this research prototype provided the learner with a command line interface for full interact.

Contents

List of Figures

List of Tables

Acknowledgements

First I would like to thank my supervisor Associate Professor Razvan Beuran, who helped me both academically and financially. Without his push and guidance, I cannot finish my master program. He was not only my supervisor but also like a friend of ours. I cannot appreciate more with such a personality as a researcher.

Many thanks to Associate Professor Shinobu Hasegawa, who helped me a lot in my research and spent a lot of time discussing with me. Without his guidance, I cannot continue my research.

I also would like to thank Professor Tadashi Matsumoto, who give me a chance to study in Japan. Thanks to Professor Brian Kurkoski, who was always willing to help me. And many thanks to my best friend Fan Zhou who shared the joys and sorrows in JAIST.

Many thanks to Dat, Long, Min, Jidong, Tan, as well as many other lab members and those who helped me during these two years. We have so many precious memories that I will keep in mind forever.

Last but not least, I want to thank my family, especially my mom, who has been keeping supporting me for the past 25 years.

Achievements

Workshop Paper

1. Zheyu Tan, Shinobu Hasegawa and Razvan Beuran (2018) "Concept Map Building from Linked Open Data for Cybersecurity Awareness Training" in 2018 83th SIG on Advanced Learning Science and Technology, SIG-ALST.

Abbreviations

LOD	Linked Open Data
RDF	Resource Description Framework
SPARQL	SPARQL Protocol and RDF Query Language
W3C	World Wide Web Consortium
HTTP	HyperText Transfer Protocol
URI	Uniform Resource Identifier
SWEO	Semantic Web Education and Outreach
PR	PageRank
SKOS	Simple Knowledge Organization System
IRT	Item response theory
CompTIA	Computing Technology Industry Association
LMS	Learning Management Systems

Chapter 1

Introduction

1.1 Motivation

Along with the rapid development of technologies, people are getting more and more cyber-related, and their daily life is exposed to kinds of cyber-attacks, such as password attacks, malware attacks, phishing emails and so on. Cybersecurity is no longer an issue discussed only between the professionals or technologists, but it is also closely related to ordinary people. In common sense, cybersecurity which we often talked about is focused on the technologies to combat cybersecurity attacks or threats, especially hardware and software. We often ignored the importance of the human factor.

In a survey carried out in 2016 by MediaPRO (specialized in cybersecurity and data privacy employee awareness programs) revealed that 88% US employees lack the awareness that needed to prevent common cyber incidents[1]. It is urgent to improve people's cybersecurity awareness, and it has significant meaning to develop a cybersecurity awareness training. Cybersecurity awareness training can help people to have solid and basic understanding of the security concepts and necessary policies. Cybersecurity awareness training is one of the most effective methods to reduce potential cybersecurity attacks in sensitive information process and organization information system protection. There are many existing cybersecurity training programmes in the world.

Nowadays, a considerable amount of money spent on cybersecurity to protect people, company and organization from cyber attacks. Expensive and sophisticated systems won't play much good without the human factor. Human is the major vulnerability in cybersecurity. Wombat Security Technologies conducted a survey revealed that almost a third of employees don't know what phishing is. Even worse is that ransomware is an unknown con-

cept to nearly two-thirds of employees.Almost 95% of cybersecurity attacks are due to human error. As much as 54% of companies have experienced one or more attacks in the last 12 months, and the number rises every month[2].

For adaptive cybersecurity awareness training, three issues can't be neglected. Firstly, what concepts we should give people to do awareness training and the relevance of the concepts used to train people. Secondly, the training contents should be updated timely. Last but not least, how to teach people efficiently and targetly. In this research, we aimed at solving these three problems.

We could dynamically get much related, and timely updated cybersecurity materials form it. This research employs the Page Rank Algorithm to calculate the importance of each concept node on the concept map to conduct adaptive awareness training later. The data nodes have higher importance have higher priority for cybersecurity awareness training.

1.2 Contribution of this thesis

In this research, we proposed a way to conduct concept map to determine what concepts we should give to people. The training contents should be updated timely.The cybersecurity awareness training in this research can be conducted very quickly and easily, and anyone can be trained at any time, for example when people assigned to a new position in the organization or faces a new challenge that needs cybersecurity knowledge. The cybersecurity awareness training in this research can be economy practiced, no room setting and less human resources needed.

In this research, we have these three major contributions:

I. We propose a way to build concept map timely updated from the Linked Open Data (LOD) database DBpedia and extracted sub concept map from it for adaptive training.

II. We proposed ways to process the built concept map. We employ the PageRank algorithm to calculate the importance of each concept node and filter algorithm to filter the irrelevant and no definition concept nodes on the concept map and use it for adaptive awareness training later.

III. We proposed a simple way to do adaptive training and implemented an adaptive awareness training system prototype. The processed concept map combined with the simple learner model provided the idea of the adaptive training. Question creation and text processing made training system into actual practice.

Chapter 2

Research Background

In this chapter, the necessary background knowledge for this thesis is introduced. At first, a brief history and some basic knowledge of LOD is given in Section 2.1 and DBpedia will be introduced in Section 2.2. After that, we will introduce the data model, RDF in Section 2.3, for describing things and the relationships between them. Then, we will talk about querying language: SPARQL in Section 2.4 and PageRank algorithm in Section 2.5. Adaptively learning theory will be introduced in Section 2.6.

2.1 Linked open data

Word Wide Web was first invented by Tim Burners-Lee in 1989[3]and it was defined as a system of interlinked hypertext documents that runs over the Internet. Web 1.0 includes three main web protocols: HTML for document formatting, HTTP for document accessing and URI for document naming. Web 1.0 was considered as a "read-only" Web since the user has rarely interaction with the website which is one of the limitations of Web 1.0. And another important limitation is that Web 1.0 pages can only be understood by people and there is no dynamic representation. The term Web 2.0 was first introduced by Darcy DiNucci in 1999 and made popular by Tim O'Reilly in late 2004[4]. Web 2.0 was defined as a platform where ordinary users can meet, collaborate, and share by using social software applications, such as Skype, Flickr, YouTube and so on. Web 2.0 was a version of Social Web while Web 3.0 was a version of Semantic Web[5]. The term "Semantic Web" refers to W3C's vision of the Web of Linked Data. Linked Data can be used for sharing machine-readable interlinked data on the Web, making data understandable to humans but also machines. Berners-Lee founded the W3C to oversee kinds of standards, and the Semantic Web is also built on these

W3C standards: the RDF data model, the SPARQL query language, the RDF Schema and OWL standards for storing vocabularies and ontologies. Berners-Lee introduced a couple of rules which known as the "Linked Data principles" in 2006[6][7]:

- 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.

Linked Open Data is a blend of Linked Data and Open Data which is linked and uses open sources. It breaks down the barriers between different data format and sources. W3C is extending the Web by publishing open data as RDF and having RDF links between various data. There are more than 31 billion RDF triples and thousands of datasets on the Web.

2.2 DBpedia

Wikipedia is a free encyclopedia and is heavily visited, moreover, it is under constant revision. DBpedia is one of the most meaningful Semantic Web projects and the most popular open knowledge dataset. Wikipedia is available in more than 250 languages, and the English version contains more than 5.79 million articles[8]. But Wikipedia has the same issues, like many other web applications, which is limited to the full-text search, with limited access to the Wikipedia knowledge[9]. DBpedia extract structured data from the information created in the Wikipedia project[8]. In this way, users can sophisticate semantically query information from it, especially relationships and properties of information. DBpedia is one of the most famous parts of the decentralized Linked Data effort.

The DBpedia dataset represents 4.58 million entities, 29.8 million links to external web pages. Thus it has a wide topic coverage. Also, DBpedia is possible of getting much further information since it is interlinked with other kinds of open datasets and it contains around 50 million links to other RDF datasets, 80.9 million links to Wikipedia categories, and 41.2 million YAGO2 categories. DBpedia uses the Resource Description Framework (RDF) to represent extracted structured information and 3 billion RDF triples included, of which 580 million were extracted from the English edition of Wikipedia and the left 2.46 billion from other language editions[8].

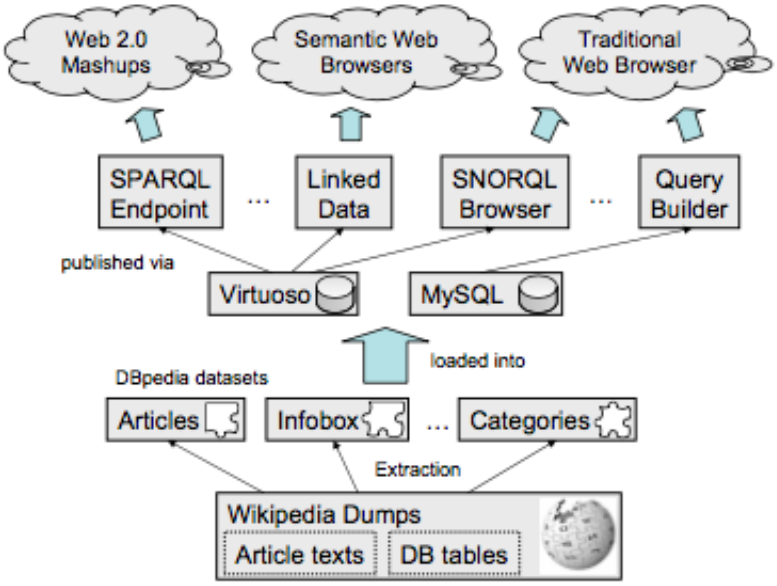

Figure 2.1: Overview of the DBpedia components

DBpedia community develop an information extraction framework, consisted of extraction, clustering, uncertainty management, and query handling. Fig. 2.1 gives an overview of the DBpedia componts.

Under the effort of W3C Semantic Web Education and Outreach (SWEO) interest group, the DBpedia interlinked to many other massive datasets and ontologies. Fig. 2.2 gives an overview of the interlinked dataset with DBpedia. DBpedia is the nucleus for the linked open data.

To provide cybersecurity training, DBpedia is a good start for looking training contents.

2.3 Resource description framework

Resource Description Framework (RDF) was originally introduced as a data model and published as a W3C recommendation in 1999[10]. RDF provides a general framework for expressing resources. The resource can be anything that has a unique identifier (URI), range from documents, physical objects to abstract concepts. RDF express data in triples: subject, predicate, object. The subject and object represent two related resources, and the predicate represents the relationship between them. This triple has a direction from

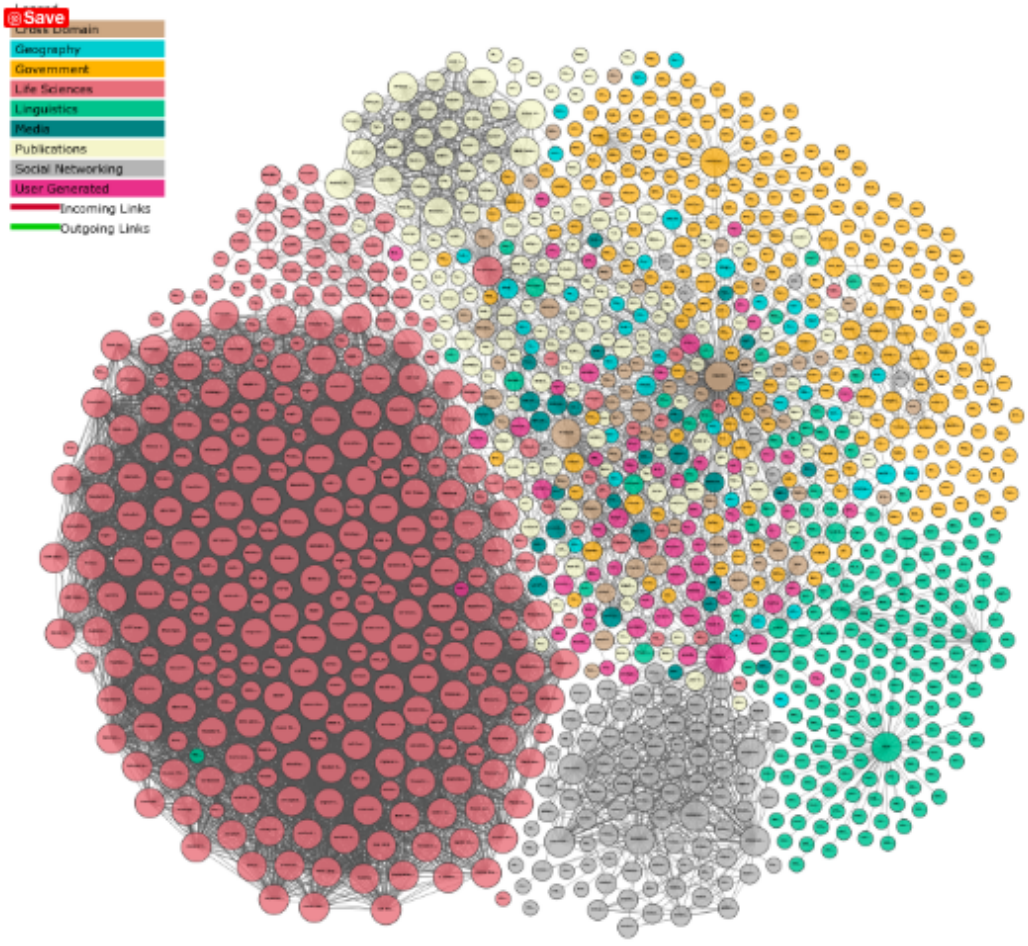

Figure 2.2: DBpedia interlinked with other LOD databases

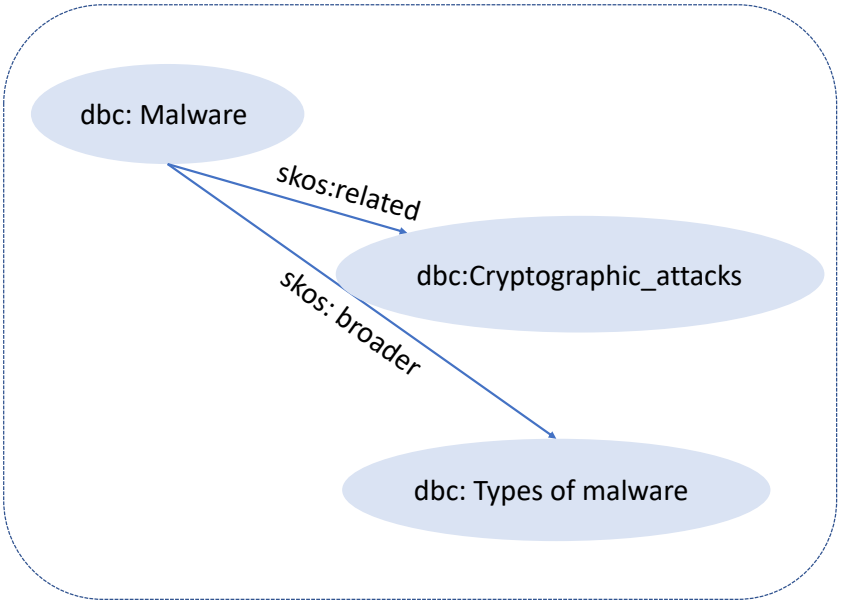

Figure 2.3: An example of RDF triples

subject to object, and the predicate is also called as property. In short, RDF is a directed, labeled graph format model representing information[11][12]

In Fig. 2.3, which contains one object $dbc : Malware$, two predicts $skos : related$ and $skos : broder$, two subjects $dbc : Cryptographic\_attacks$ and $dbc : Types\ of\ malware$. An RDF statement is a directed graph from subject to object. Fig. 2.3 contains two triples. Each triple is a fact.

RDF describes resources in classes, properties, and values. Also, RDF also needs a way to define application-specific classes and properties, which are must be defined using extensions to RDF. RDF Schema also called RDF vocabularies, is a set of classes with certain properties using the RDF extensible knowledge representation data model, providing basic elements for the description of ontologies and structure RDF resources[13].

2.4 SPARQL

SPARQL is a standard language to query graph data expressed as RDF triples. It is one of three core standards of the Semantic Web, along with RDF and OWL[14]. SPARQL is capable of querying required and optional graph patterns with specifying conjunctions and disjunctions. Usually, SPARQL query is a set of patterns called basic graph pattern with the subject, predicate or object may be a variable, and the result of a SPARQL query is a solution sequence. In other words, querying data by SPARQL is a process of finding certain graphs that match required graph patterns. There are four kinds of query form, SELECT, CONSTRUCT, DESCRIBE and ASK, and multiple solution sequence modifiers, such as LIMIT, ORDER BY, OFFSET and so on. By the flexibility using of query form, modifiers plus some operators and OPTIONAL value, we can query as we want[15]. Fig. 2.4 is an example of using SPARQL to query. First three lines are declaring prefixes. For example, prefix tells us prefix $rdfs$ will stand for the URI: `/http://www.w3.org/2000/01/rdf-schema#/` instead of writing out the full URIs every time. $SELECT$ query form will return a table with four columns for four variables: child1, child2, child1label, and child2label. $WHERE$ specifies basic graph pattern to match against the data graph. $FILTER$ constraint solutions specified that we only want English language labels. $BIND$ assign the result concept1name to a variable concept1lable. Querying results showed in Fig. 2.5

2.5 Page rank algorithm

Page Rank algorithm is one of the core methods that Google uses to determine the relevance or importance of a page. Page Rank algorithm is defined as follows[16]:

We assume page A has pages T1...Tn which point to it (i.e., are citations). The parameter d is a damping factor which can be set between 0 and 1. We usually set d to 0.85. Also, C(A) is defined as the number of links going out of page A. The PageRank of page A is given as follows:

$$PR(A) = (1-d) + d(\frac{PR(T_1)}{C(T_1)} + ... + \frac{PR(T_n)}{C(T_n)}) \tag{2.1}$$

Lawrence Page and Sergey Brin give an Intuitive Justification in their published paper. They consider PageRank as a model of user behavior that user random suffer the Internet. The users visit a page with a certain probability is its PageRank. This probability is given by the number of links on

Figure 2.4: An example of SPARQL query: query children of the Computer security

child	childlabel
http://dbpedia.org/resource/Category:Access_control	"Access control"@en
http://dbpedia.org/resource/Category:Computer_access_control	"Computer access control"@en
http://dbpedia.org/resource/Category:Computer_security_standards	"Computer security standards"@en
http://dbpedia.org/resource/Category:Operating_system_security	"Operating system security"@en
http://dbpedia.org/resource/Category:Trusted_computing	"Trusted computing"@en
http://dbpedia.org/resource/Category:Information_privacy	"Information privacy"@en
http://dbpedia.org/resource/Category:Computer_security_companies	"Computer security companies"@en
http://dbpedia.org/resource/Category:IT_risk_management	"IT risk management"@en
http://dbpedia.org/resource/Category:Computer_network_security	"Computer network security"@en
http://dbpedia.org/resource/Category:Computer_security_exploits	"Computer security exploits"@en
http://dbpedia.org/resource/Category:Computer_security_organizations	"Computer security organizations"@en
http://dbpedia.org/resource/Category:Computer_security_procedures	"Computer security procedures"@en
http://dbpedia.org/resource/Category:Cryptography	"Cryptography"@en
http://dbpedia.org/resource/Category:Data_security	"Data security"@en
http://dbpedia.org/resource/Category:People_associated_with_computer_security	"People associated with computer security"@en
http://dbpedia.org/resource/Category:Mobile_security	"Mobile security"@en
http://dbpedia.org/resource/Category:Computer_forensics	"Computer forensics"@en
http://dbpedia.org/resource/Category:Computer_security_software	"Computer security software"@en
http://dbpedia.org/resource/Category:Computer_surveillance	"Computer surveillance"@en
http://dbpedia.org/resource/Category:Computer_security_books	"Computer security books"@en
http://dbpedia.org/resource/Category:Computer_security_models	"Computer security models"@en
http://dbpedia.org/resource/Category:Computer_security_qualifications	"Computer security qualifications"@en

Figure 2.5: The result of the SPARQL query in Fig. 2.4

that page, and the PageRank of a page is divided by the number of links on that page. The damping factor d is the probability that the random surfer is jumping to another random page at each page, so the probability is represented as a constant $(1-d)$ in the above definition. There is another version of the Page Rank algorithm that Lawrence Page and Sergey Brin published in another paper. Page Rank algorithm is defined as follows[16]:

$$PR(A) = \frac{(1-d)}{N} + d(\frac{PR(T_1)}{C(T_1)} + ... + \frac{PR(T_n)}{C(T_n)}) \tag{2.2}$$

In this second version, the probability of random surfer a page is weighed by the total number of web pages N. The page's PageRank is the actual probability for a random surfer reaching that page after clicking many links. Then the PageRanks form a probability distribution over the all web pages, and the sum of all page's PageRank will be one. These two versions have no fundamental difference.

2.6 Adaptive learning theory

Adaptive learning is intelligent which interacted with learners by using computer technology. Computers adapt training questions according to the need for a learner, or from training feedback of learners. Cybersecurity training will be improved when training is personalized which adapted to each learner[17]. Tradition adaptive learning system usually divided into 4 models[17]:

I. Expert model

   The expert model contains training materials that used to teach student. It can be questions and solutions, or lessons and tutorials.

II. Learner model

   The Learner model (also refered as Student model) contain kinds information of the learners, such as domain knowledge, learning performance, interests, preference, goal, tasks, background and so on.

III. Instructive model

   The Instructive model combine the previous two models together to show next cobntent

IV. Instrucrional environment

   The Instrucrional environment is user interface or training platfrom

2.7 Related projects

Cybersecurity awareness of people can affect many aspects of a company, an organization, or even people's daily life. There is much current cybersecurity awareness training provided, and especially some large company has their employees to have regular cybersecurity awareness training once or twice a year. This section will introduce the related training projects to this thesis.

Currently, their multiple types of cybersecurity awareness training approach. Such as the breakroom approach, which people are gathered at the break time and are told basic tips about cybersecurity. The security video approach, which shows short cyber security training related videos to people.

iHACCO is one of high-quality online training companies and also provides cyber security awareness course. The course of iHACCO cyber security awareness is made up of simple pdf files of DON'Ts and DOs.

ESET Cybersecurity Awareness Training is one of famous training and test. ESET Cybersecurity Awareness Training provides training from 5 aspects: threats overview, password safety, internet protection, email protection, and preventive measures.

But acquiring training contents from LOD and use the querying results to generate adaptive awareness training is rare.

Chapter 3

Adaptive Security Awareness Training System

3.1 System design

The new system design is illustrated in Fig. 3.1. The main blocks in this system are presented in the following sections. Section 3.2 present how to build concept map from the LOD database DBpedia. The concept map is dynamic get much related and timely updated cybersecurity materials. Section 3.3 present the concept relevance estimation which employs the Page Rank Algorithm to calculate the importance of each concept node on the concept map to conduct adaptive awareness training later in the next several sections. Section 3.4 introduce the filtering. Section 3.5 explains how to generate questions. Learner model, update, and adaptive training will be shown in section 3.6.

Chapter 2 introduced that tradition adaptive learning system usually divided into such four models: Expert model, Learner model, Instructive model, and Instructional environment. The expert model contains training materials that used to teach the student. It can be questions and solutions, or lessons and tutorials. The question set in Section 3.5 functioned as an expert model in this research. The Learner model contains kinds of information of the learners. The learner model in Section 3.6 functioned as the learner model to improve practice training. The Instructive model combines the previous two models to show the next content. The update in Section 3.6 functioned as the instructive model in this research. The Instructional environment is a user interface or training platform. The adaptive awareness training system prototype in this research contained the command line interface for the training.

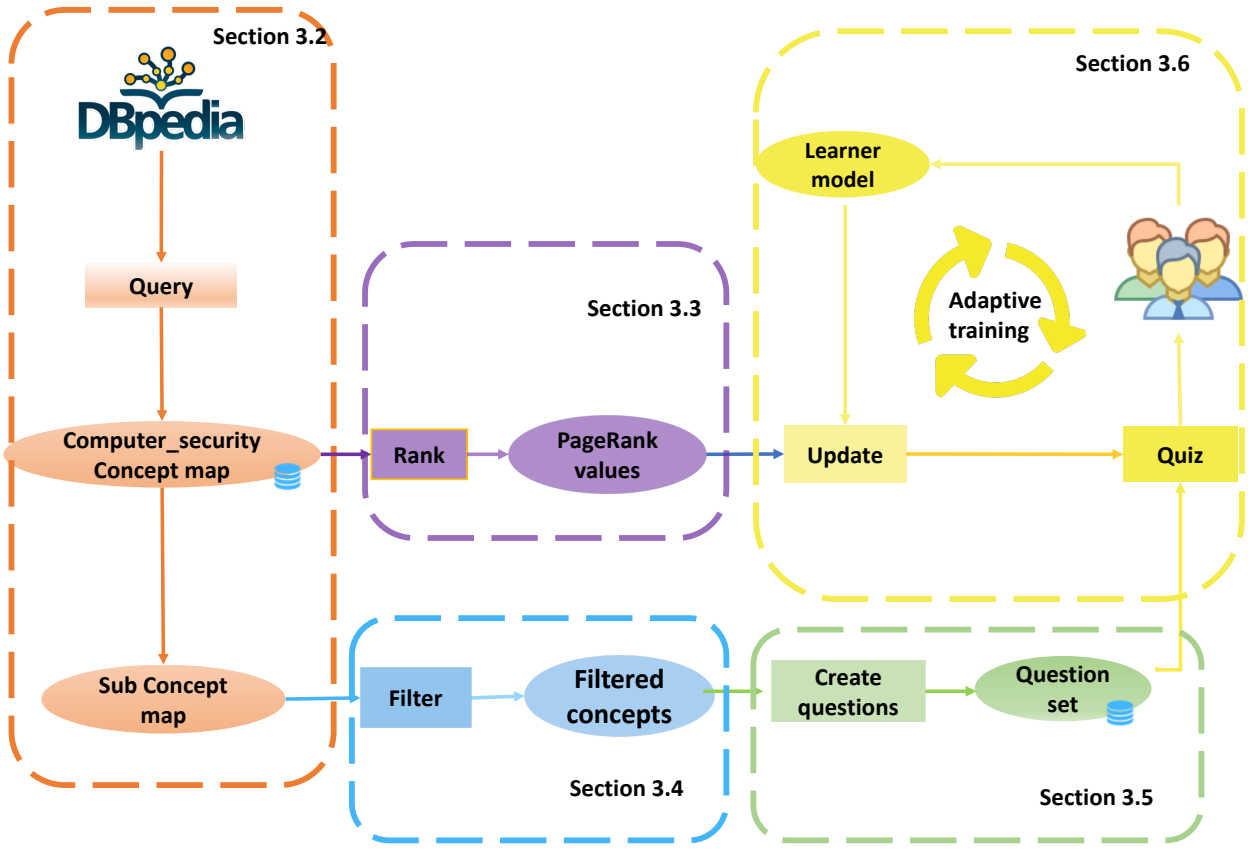

Figure 3.1: System design for adaptive security awareness training

3.2 Concept map construction

This section builds the Computer_security map. Then in Section 3.3, construct the subconcept map from the Computer_security map made in Section 3.2 based on the given keyword.

3.2.1 DBpedia query via SPARQL

DBpedia was constructed by extracting data from Wikipedia, for example, titles, page links, categories and so on. Facts in DBpedia are in RDF graph. The entire knowledge can be either download or, it can be accessed by public SPARQL endpoint. In this thesis, access by public SPARQL endpoint is chosen. This section uses SPARQL querying data from DBpedia and builds a concept map based on the given keyword.

Simple Knowledge Organization System (SKOS) is a W3C recommendation designed for representation of taxonomies classification schemes and so on. The primper emolument categories of SKOS are concepts, labels, notations, semantic relations, and collections. The main element used in this research is semantic relations. SKOS semantic relations are designed to declare relationships between concepts within a concept scheme[18].

The query in this thesis will use an important property: "skos: broad"[19]. SKOS is Simple Knowledge Organization System which is a W3C recommendation built based on RDF and RDFS. It is a part of the Semantic Web family, and its main purpose is to use such kind of vocabularies as linked data for publishing. The vocabulary of SKOS includes kinds elements that work together for representing, such as concepts, labels, relationships and so on. "skos: broad" represent hierarchical relationships between concepts and this property relates a concept to another concept that is more general in meaning. For example, "A $skos$: $broader$ B" means B has a more general and broader meaning than A. Broader concepts are typically rendered as parents in a concept hierarchy (tree). SPARQL engines can usually return results in different types, for example, XML format, JSON format, and CSV format. In this thesis, JSON format as chosen, since the libraries used to process JSON objects are available in most programming languages and it's convenient for the Page Rank Algorithm implementation in this research. The concept map is in JSON format and to frankly display it.

3.2.2 Query strategy

The query strategy is essential if we aimed at building an efficient and useful concept map for cybersecurity awareness training. The entire concept map is built based on the given keyword: Computer_security. "$A\ skos : broader\ B$" means B has a more general and broader meaning than A. Broader concepts are typically rendered as parents in a concept hierarchy (tree).

The concept map is a collection of entities called nodes, which is concepts in this research. Concepts(nodes) are connected by edges which is the property "$skos : broader$" , managed the relationship between concepts, in this research. "$rdfs : label$" is an instance of "$rdf : Property$" that used to provide a human-readable version of a resource's name[13]. The first or topmost node of the tree is the root, in this research, the root node is the keyword: Computer_security. In this research, we query the descendent concepts of the Computer_security. In computer science, there are has multiple tree traversal algorithms: Depth-first and Breadthfirst. In this research, we reach the descendent concepts of the Computer_security with Breadthfirst algorithm in level order.

The core query algorithm in the query strategy used in this research is as following:

```
SELECT DISTINCT ?child ?childlabel

WHERE{
?child skos:broader ⟨http://dbpedia.org/resource/Category:concept⟩;
rdfs:label ?childname.
FILTER (LANG(?childname) = 'en')
BIND (?childname AS ?childlabel)
}
```

The descendent concepts of Computer_security is enormous. This research shows several Computer_security maps with various depth. As depth growing, the number of nodes/concepts increase rapidly.

I. Fig. 3.2 is the concept map built based on the given keyword: Computer security. It only queries the children concepts of the keyword concept. It contains 23 concept nodes in total, in which the root is Computer_security, and it has 22 children concepts.

II. Fig. 3.3 is the concept map built based on the given keyword: Computer security. It queries the children concepts and the grandchildren concept

Figure 3.2: Concept map of Computer_security,depth = 1, 23 concept nodes inside

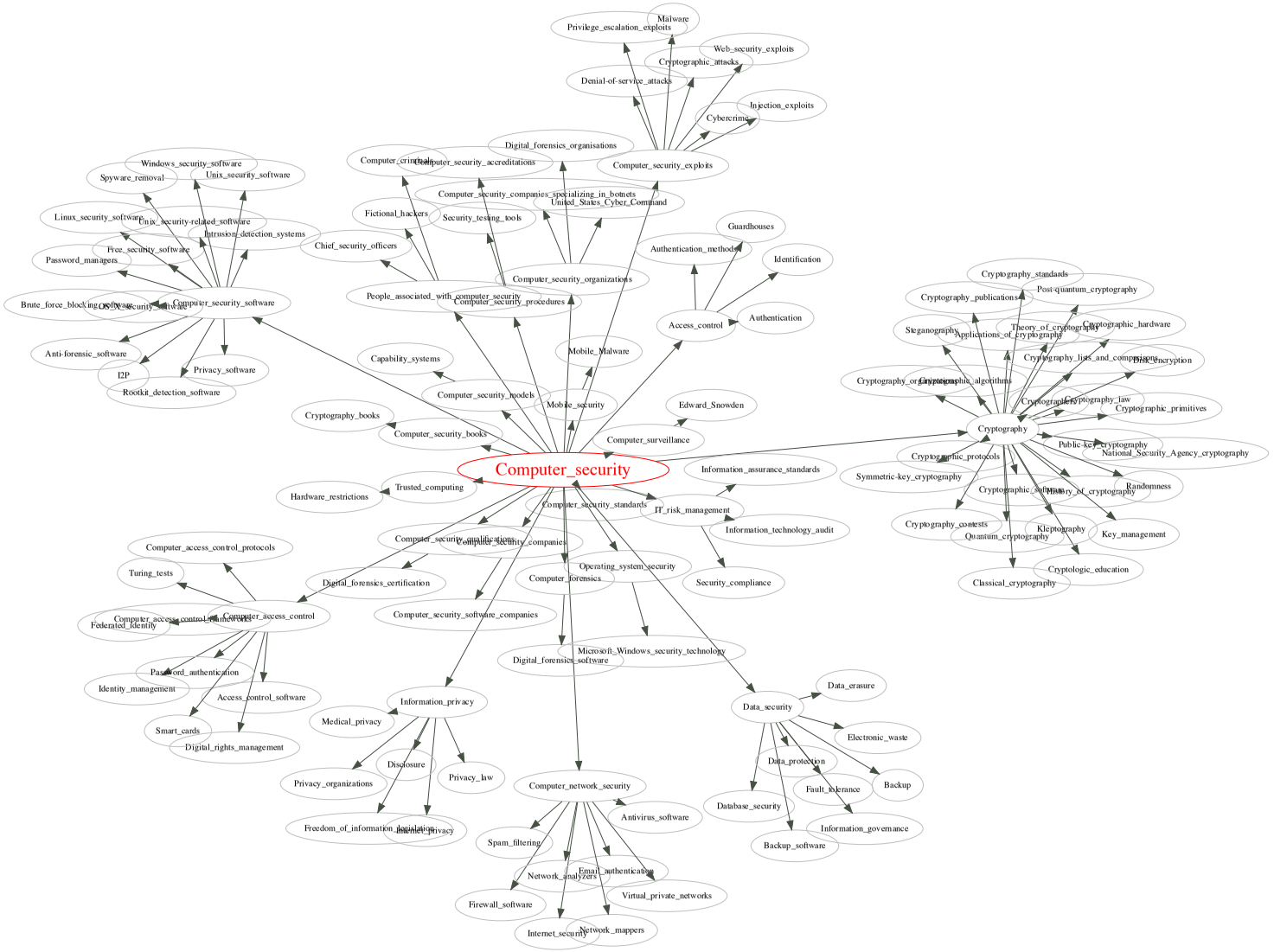

Figure 3.3: Concept map of Computer_security,depth = 2, 126 concept nodes inside

of the keyword concept. It contains 126 concept nodes in total, in which the root is Computer_security, and it has 22 children concepts, and 103 grandchildren concepts.

III. The descendent concepts of Computer_security is enormous. This Tab. 3.1 shows several Computer_security maps with various depth. As depth growing, the number of nodes/concepts increase rapidly.

Table 3.1: Concept map, keyword: Computer security

Depth	**concept nodes in each level**	**total concept nodes**
0	1	1
1	22	23
2	103	126
3	205	331
4	287	618
5	266	884
6	463	1347
7	1293	2640

3.2.3 Sub concept map generation

In Section 3.2, we uses SPARQL querying data from DBpedia and builds a concept map based on the given keyword: Computer_security. The descendent concepts of Computer_security are enormous. As depth growing, the number of nodes/concepts increase rapidly, and the time for accessing the Internet growing as the depth increased. When using this big concept map for training, we extract sub concept map form the Computer security map generated in Section 3.2.2.

Fig.3.4 is an overview of the sub concept map derived from the big one, and the keyword of this sub concept map is Malware. Malware is one of the descendant concepts of the Computer_security. The Malware concept is on level 2 of the Computer_security concept map(Computer_security is on level 0).

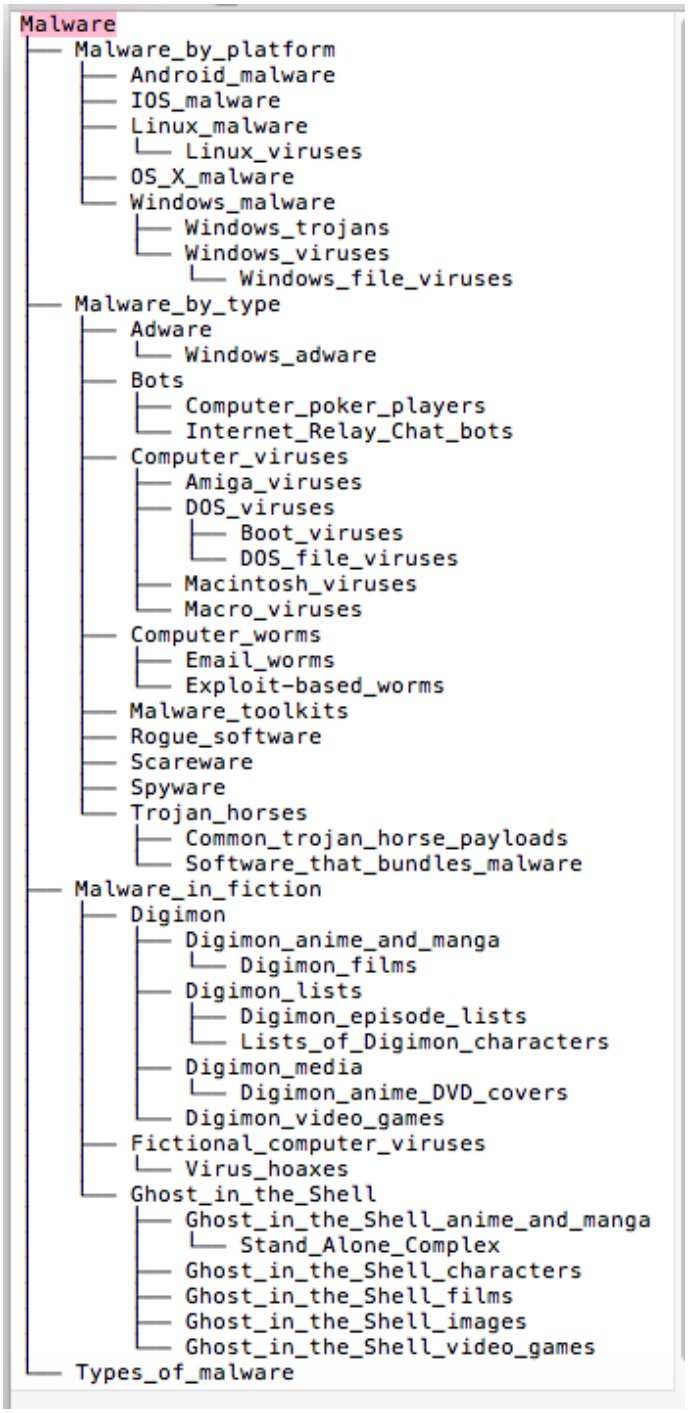

Figure 3.4: Sub concept map from Computer_security concept map,keyword:malware, 54 concept nodes inside

3.3 Concept relevance estimation

Google use Page Rank algorithm to determine the relevance or importance of a page and the importance of a page is determined by the number of links going out of this page. In this research, a concept map is a set of interlinked concept nodes. We assume that the importance of a concept node is determined by the number of linked concept nodes going out of this concept node. In Page Rank algorithm, the ranking of a page is recursively given by the ranking of those pages which linked to it. In this paper, in the same way, the ranking of a concept is recursively given by the ranking of those concepts which related/linked to it. But how do we know the final PageRank value of those concepts since the PageRank of a concept is always given recursively by the PageRank of related concepts. The answer can be found in Lawrence Page and Sergey Brin's paper . We repeat calculations many times until the values stop changing much $(P_{n+1} < -P_{n+1}) < \epsilon$. In this paper, we guess PagePanks of concepts at first and it doesn't matter where you start. When the PagePank calculations converged or reached a fixed point, the normalized probability distribution will be 1.0. According to Lawrence Page and Sergey Brin, around 100 iterations are necessary to get a good approximation of the PageRank values of the whole web.

We constructed the hierarchy concept map from the Categories in Section 3.2. The sub concept map generated from the Computer_security map may still be large, for example, the Malware sub concept map in Fig. 3.4 contains 54 concept nodes which may not all useful for the training later. Even they are all useful ideally; we still need to decide the concepts which is more important to Malware adaptive training. So before using this concept map for adaptive training, we need to process this concept map first. And the first step to process the concept map is to estimates the concept importance by applying PageRank algorithm on the Computer_security map. Tab. 3.2 is the part of final calculated importance/PageRank value on Computer_security map after 56 iterations. The damping factor is set to 0.85, the maximum number of iteration is set to 100, the epsilon is set to 10^{-9}.

Table 3.2: Final PageRanks, Keyword: Computer security, iterations: 56

Concept	**PageRanks**
computer security	0.10382910
access control	0.05869410
computer access control	0.00354294
computer security standards	0.00011901
operating system security	0.00022017
trusted computing	0.00183606
information privacy	0.00642375
computer security companies	0.00030615
it risk management	0.00042248
computer network security	0.00240053
computer security exploits	0.00679187
computer security organizations	0.00042248
computer security procedures	0.00032132
cryptography	0.02845521
data security	0.00668850
people associated with computer security'	0.00208197
mobile security	0.00022017
computer forensics	0.00022017
computer security software	0.00187915
computer surveillance	0.00030615
computer security books	0.00022017

3.4 Filtering

In section 3.2, we constructed the concept map. In section3.3, we processed the concept map by applying PageRank algorithm to calculate the importance of each node. In this part, we will discuss the second step of processing the concept map: filtering, since not every concept in the sub concept map is useful. We need to filter irrelevant and no definition concepts.

3.4.1 Filtering concepts based on concepts utility

In the previous concept map, not every concept has definition in DBpedia. On the hypothesis that concept without definition is not essential for practical training. This paper discards no definition concept. At the beginning of this research, we directly check each concept has a definition or not in practice. The core query algorithm is that we filter no definition at the beginning of this research as follows:

```
SELECT DISTINCT ?definition

WHERE{
⟨http://dbpedia.org/resource/concept⟩ dbo:abstract ?definition
FILTER (LANG(?definition) = 'en')
}
```

Table 3.3: The first filtered concepts

Concept
Malware
Spyware
Ransomware
Scareware

Combine with Section 3.4.2, after filtered irrelevant concepts, only four concept questions left, showed in Tab. 3.3. In this section, we checked these four concepts. Even all these concepts have a definition, but the left number of concept related to Malware is unsatisfactory. And in the late of this research, when we do the exact example with Computer_worms, we found

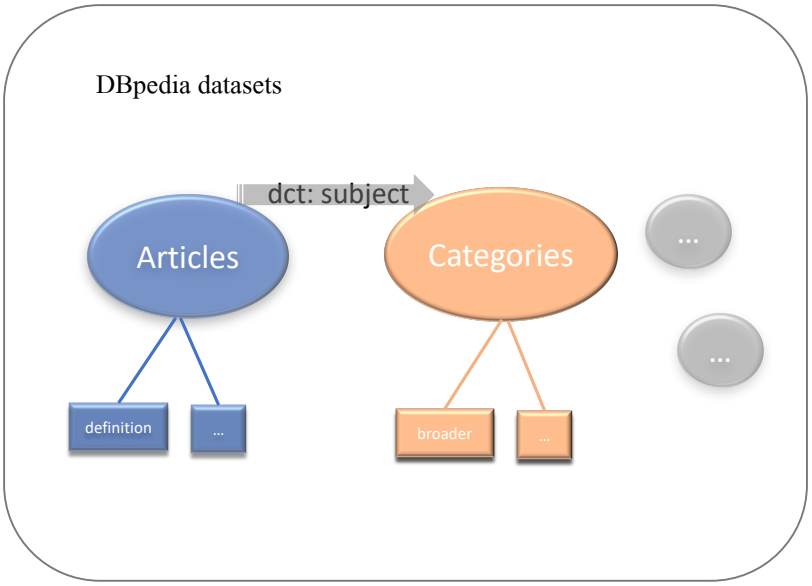

Figure 3.5: Link Articles with Categories

out that the Computer_worms does not have a definition, but the Computer_worm does have a definition. Without any doubt, no matter Computer_worm or Computer_worms is useful for awareness training. This unexpected result led us to think about the reason for this situation and looking for a more advanced filter algorithm.

In Section 2.2, Fig. 2.1 gives an overview of the DBpedia information extraction techniques. In our initial research, we ignored the connection or convert from Categories to Articles. We use Computer_worms to query the definition in Articles instead of the counter of Computer_worms in Articles.

We constructed the hierarchy concept map from the Categories and the definition in the Articles. Before we query the definition in the Articles, we first link Categories with Articles. Fig.3.5 given the DBpedia datasets and show the idea of how to connect Categories and Articles by use the property of "$dct : subject$".

Part of the core query algorithm in advanced filter of this research as follows:

```
SELECT DISTINCT ?child_defi ?childlabel_defi

WHERE{
?child_defi dct:subject ⟨http://dbpedia.org/resource/Category:concept⟩ .
FILTER (LANG(?childname_defi) = 'en')
BIND (?childname_defi AS ?childlabel_defi) }
```

The link from Articles to Categories is represented as $dct : subject$, and from Categories to their superordinates as $skos : broader$. For the concept Malware in Categories, it has multiple results in Articles. The results is in the Fig.3.6. Fig.3.7 shows the results of link from Categories to Articles for the keyword: Computer_worms. In this research, among the multiple results of $dct : subject$, using a simple Nature Language Processing library -fuzzywuzzy to find the most approximate string matching of the keyword. In Fig.3.6, the most approximate matching of the Malware is $Malware$.In Fig.3.7, the most approximate matching of the Computer_worms is $Computer\_worm$. Matching in each figure is highlighted in red.

child_defi	childlabel_defi
http://dbpedia.org/resource/Download.com	"Download.com"@en
http://dbpedia.org/resource/Mahdi_(malware)	"Mahdi (malware)"@en
http://dbpedia.org/resource/SpySheriff	"SpySheriff"@en
http://dbpedia.org/resource/Malware_analysis	"Malware analysis"@en
http://dbpedia.org/resource/Regin_(malware)	"Regin (malware)"@en
http://dbpedia.org/resource/China_Internet_Network_Information_Center	"China Internet Network Information Center"@en
http://dbpedia.org/resource/Whitelist	"Whitelist"@en
http://dbpedia.org/resource/Underhanded_C_Contest	"Underhanded C Contest"@en
http://dbpedia.org/resource/Timeline_of_computer_viruses_and_worms	"Timeline of computer viruses and worms"@en
http://dbpedia.org/resource/Download.ject	"Download.ject"@en
http://dbpedia.org/resource/Stuxnet	"Stuxnet"@en
http://dbpedia.org/resource/Genieo	"Genieo"@en
http://dbpedia.org/resource/NBName	"NBName"@en
http://dbpedia.org/resource/Moralityware	"Moralityware"@en
http://dbpedia.org/resource/Wirelurker	"Wirelurker"@en
http://dbpedia.org/resource/Malware	"Malware"@en
http://dbpedia.org/resource/Potentially_unwanted_program	"Potentially unwanted program"@en
http://dbpedia.org/resource/Tribe_Flood_Network	"Tribe Flood Network"@en
http://dbpedia.org/resource/Beast_(Trojan_horse)	"Beast (Trojan horse)"@en
http://dbpedia.org/resource/Blackshades	"Blackshades"@en
http://dbpedia.org/resource/Dridex	"Dridex"@en
http://dbpedia.org/resource/Claria_Corporation	"Claria Corporation"@en
http://dbpedia.org/resource/Yahoo!_Assistant	"Yahoo! Assistant"@en
http://dbpedia.org/resource/Carbanak	"Carbanak"@en
http://dbpedia.org/resource/Locky	"Locky"@en
http://dbpedia.org/resource/CDP_Spoofing	"CDP Spoofing"@en

Figure 3.6: Results of link Categories with Articles, keyword: Malware

child_defi	childlabel_defi
http://dbpedia.org/resource/WANK_(computer_worm)	"WANK (computer worm)"@en
http://dbpedia.org/resource/Alcra_(computer_worm)	"Alcra (computer worm)"@en
http://dbpedia.org/resource/W32.Gammima.AG	"W32.Gammima.AG"@en
http://dbpedia.org/resource/Warhol_worm	"Warhol worm"@en
http://dbpedia.org/resource/Agobot	"Agobot"@en
http://dbpedia.org/resource/Kak_worm	"Kak worm"@en
http://dbpedia.org/resource/Vundo	"Vundo"@en
http://dbpedia.org/resource/Pikachu_virus	"Pikachu virus"@en
http://dbpedia.org/resource/Computer_worm	"Computer worm"@en
http://dbpedia.org/resource/Spybot_worm	"Spybot worm"@en
http://dbpedia.org/resource/Kama_Sutra_(computer_worm)	"Kama Sutra (computer worm)"@en
http://dbpedia.org/resource/Mikeyy	"Mikeyy"@en
http://dbpedia.org/resource/Santy	"Santy"@en
http://dbpedia.org/resource/Conficker	"Conficker"@en
http://dbpedia.org/resource/Father_Christmas_(computer_worm)	"Father Christmas (computer worm)"@en
http://dbpedia.org/resource/Samy_(computer_worm)	"Samy (computer worm)"@en
http://dbpedia.org/resource/Blackworm	"Blackworm"@en
http://dbpedia.org/resource/Leap_(computer_worm)	"Leap (computer worm)"@en
http://dbpedia.org/resource/Redesi	"Redesi"@en
http://dbpedia.org/resource/Anna_Kournikova_(computer_virus)	"Anna Kournikova (computer virus)"@en
http://dbpedia.org/resource/Morris_worm	"Morris worm"@en
http://dbpedia.org/resource/Voyager_(computer_worm)	"Voyager (computer worm)"@en
http://dbpedia.org/resource/Gruel_(computer_worm)	"Gruel (computer worm)"@en
http://dbpedia.org/resource/BuluBebek	"BuluBebek"@en
http://dbpedia.org/resource/Devnull	"Devnull"@en
http://dbpedia.org/resource/Sobig	"Sobig"@en

Figure 3.7: Results of link Categories with Articles, keyword: Computer_worms

3.4.2 Filtering concepts based on DBpedia class and property

There is another case needed to be considered, that is, not every concept is relevant to practical training. For example, concept “Digimon” is in the concept map built from the keyword “Malware” (“Digimon” is the grandchildren concept of “Malware”). But it is an instance of a “Game” class, it is irrelevant to the cybersecurity awareness training. We filter this kind of concept based on DBpedia class and property. This thesis discards irrelevant concepts. Tab. 3.4 is the list of irrelevant classes used for filtering in thesis. Tab. 3.5 is the final filtered irrelevant concepts for ”Keyword: Malware”

Table 3.4: List of irrelevant classes used for filtering

Irrelevant classes
movie
Movie CW
televisionshow
animal
grape
place
Planet
Location
Agent
Q386724
game

After linked and filtered irrelevant concepts, 20 concepts related to Malware left in Tab. 3.4. Comparing with Fig. 3.4 that the concepts in Malware sub concept map *level* 1 all filtered out since those four concepts have no definition. The *Digimon* concept also filtered out since it is irrelevant to the keyword Malware.

Table 3.5: The second filtered concepts after link Categories with Articles

concept	**pageRank**	**level**
Malware	0.00309623	0
Computer_virus	0.00050839	2
Winwebsec	0.00029768	2
Computer_worm	0.00023484	2
AIDS_(Trojan_horse)	0.00023484	2
Linux_malware	0.00016091	2
Dendroid_(malware)	8.69785912e-05	2
Blackhole_exploit_kit	8.69785912e-05	2
Rogue_security_software	8.69785912e-05	2
Scareware	8.69785912e-05	2
Spyware	8.69785912e-05	2
Bliss_(virus)	8.69785912e-05	3
Zlob_trojan	8.69785912e-05	3
Computer_poker_players	8.69785912e-05	3
SCA_(computer_virus)	8.69785912e-05	3
Melissa_(computer_virus)	8.69785912e-05	3
Mimail	8.69785912e-05	3
Blaster_(computer_worm)	8.69785912e-05	3
Back_Orifice	8.69785912e-05	3
Virus_hoax	8.69785912e-05	3

3.5 Generating automatically questions

Section 3.4 processed the sub concept map with two steps filtering and in this section, using the filtered concept list to generate questions. Generating question using the definition queried from Articles. The generated question for each concept will be stored in the JSON format as follows:

```
question =
{
'id': concept,
'body': 'What is ' + concept ? ,
'choices': choices [incorrect1, incorrect2, incorrect3, correct],
'ans': ans
}
```

This research provided multiple-choice questions. Multiple choices question which has multiple related competitive incorrect answer will not only help the learner to learn but also will make the learning more interesting.

The stem of the question is in a straight form of *What is concept*?, where the *concept* is the given keyword from the learner.

The correct answer to the question is coming from the concept itself.

The incorrect answers to the question are coming from the concepts that on the same level of the keyword concept on the Computer_security tree.

3.5.1 Definition and text processing

In Section 3.2 we built the big Computer_securtity map and extracted the sub concept map on the given keyword considering the learner need. In Section 3.3, we processed the big Computer_securtity map by employing the PageRank algorithm on it. In Section 3.3, to make the concept map practical useful for actual training, we filtered the concept map. Now we have filtered left concepts in Tab. 3.5. This section will show how to generate each part of the question by retrieval the definition from DBpedia dataset and text processed the definition materials.

Get the definition of the concepts in the filtered concept list first, neither correct answer or incorrect answer is coming from the definition (from the concept itself or the irrelevant concept). Part of the core query algorithm is querying the definition of this research as follows:

SELECT DISTINCT ?definition

WHERE{
⟨`http://dbpedia.org/resource/`*concept*⟩ dbo:abstract ?definition.
FILTER (LANG(?definition) = 'en') }

The idea to generate incorrect choices and correct choices from definition materials is the same. Both choices are in a straight form of "...*is concept definition*.", replace the keyword concept in concept definition with "...".

Replacement using python regular expression operations. In this research, if the definition text of the concept is too long, extracted the first sentence from it and replaced the keyword concept. In the future practical training, when handle the special case, liking concept keyword in the definition is very different from itself, need particular carefully processed.

3.6 Adaptive security awareness training

This part present Learner model which tracks the learner's understanding of the concept. And introduced the related update algorithm.

The training in this research based on the given keyword and aimed to let the learner handle all the training concepts related to the given keyword. Training consists of multiple small quizzes, the number of quizzes is determined by the learner's knowledge and the sub concept map.

Based on the given keyword, the size of the generated sub concept map varied. For example, the 'Spyware' is the leaf node of the Computer_security map, so the size of the 'Spyware' sub concept map is small, only one concept inside; the 'Malware' sub concept map includes 20 useful concepts inside. In this adaptive security awareness training system, training all 20 useful concepts about Malware is time-consuming. So this research set a threshold of the number of the training concepts to six, this threshold can be modified easily.

3.6.1 Learner model

The learner model present in this section using a straightforward version to interact with the learner and adaptive awareness learning system. The learner model reveals the understandings and misunderstanding to correct knowledge.

Tab.3.7 $\sim$ Tab.3.10 are the four learner models for an actual pratical training combined update algorithm in next section.

Table 3.6: Selected training concepts related to the Malware

concept	**pageRank**	**level**
Malware	0.00309623	0
Computer_virus	0.00050839	2
Winwebsec	0.00029768	2
Computer_worm	0.00023484	2
AIDS_(Trojan_horse)	0.00023484	2
Linux_malware	0.00016091	2

3.6.2 Update algorithm

Update algorithm bring concept map and learner model together to determine which training content should be given next. In this research, adaptive awareness training contains several small quizzes. Each quiz consists of four questions — the number of quizzes determined by the size of the subconcept map and the knowledge of the learner.

In the generated sub concept map, it provides useful information which is the level of concept.

1. First quiz: The threshold for the number of training concepts is six which means that in the training initialization, six concepts should be selected. Traverse the sub concept map level by level. In each level, the concept with higher PageRank is the priority to be selected. Tab. 3.6 showed the selected training concepts related to the Malware.

2. Learner model update: Update the learner model based on the result of each quiz.

3. Generate the next quiz: Select training concepts based on the learner model, the correctly answered concept should not appear in the next quiz. If the number of unhandled concepts is larger than four, traverse the sub concept map level by level, and in each level, the concept with higher PageRank is priority to be selected.

Table 3.7: Leaerner model for quiz 1

concept	**learner understanding**
Malware	✓
Computer_virus	✓
Winwebsec	×
Computer_worm	×

Table 3.8: Learner model for quiz 2

concept	**learner understanding**
Winwebsec	✓
Computer_worm	✓
AIDS_(Trojan_horse)	×
Linux_malware	×

Table 3.9: Learner model for quiz 3

concept	**learner understanding**
AIDS_(Trojan_horse)	✓
Linux_malware	×

Table 3.10: Learner model for quiz 4

concept	**learner understanding**
Linux_malware	✓

Take Malware training as an example, ask a learner do this training. This learner has a little background of cybersecurity. This learner finishes the training in four quizzes. Fig.3.8 shows the first quiz in adaptive training. Tab.3.7 $\sim$ Tab.3.10 is the learner model for each quiz.

We decided to use the Moodle LMS for visualizing and taking the quiz to have a good user experience. Convert JSON format questions to YAML format. Then use CyLMS convert yaml file to SCORM format, and then displayed in Moodle (Fig.3.9). CyLMS is for training content management. And is a part of integrated cybersecurity training framework CyTrONE. Moodle is a free and open-source learning management system. Fig.3.10 illustrated the generated question displyed in Moodle.

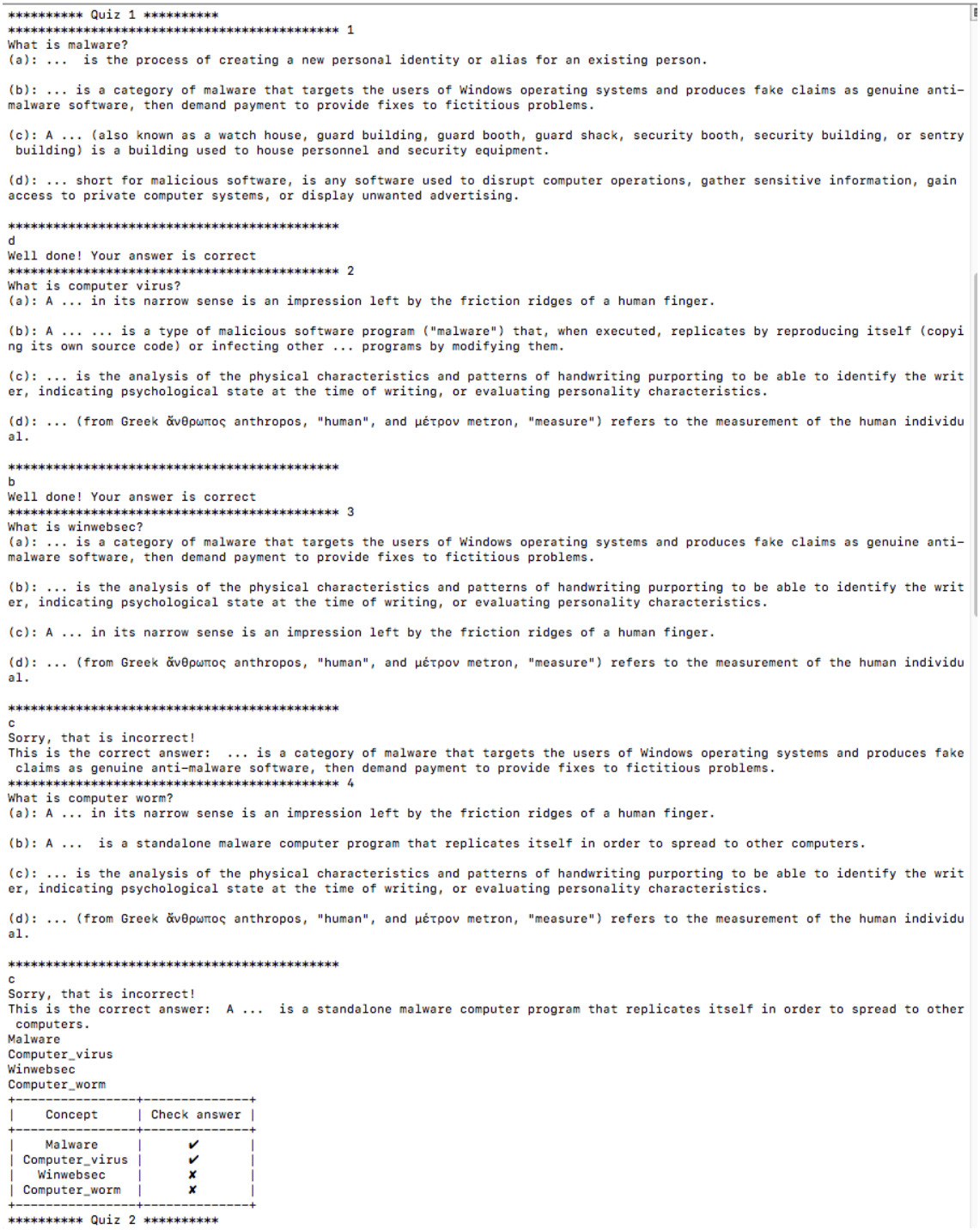

Figure 3.8: The first quiz iin Malware training

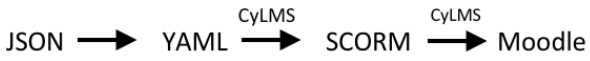

Figure 3.9: How JSON format question convert into Moodle file

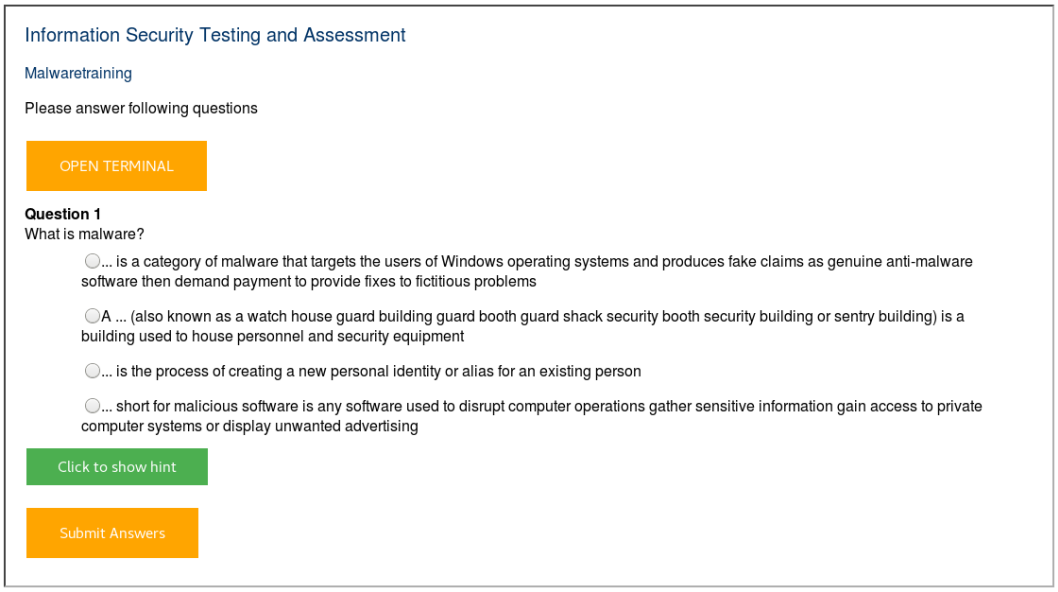

Figure 3.10: Question displayed in Moodle

Chapter 4

System Evaluation

4.1 Concept map evaluation

This research, in Chapter 3, uses SPARQL querying data from DBpedia and builds a concept map based on the given keyword: Computer_security. The descendent concepts of Computer_security are enormous. As depth growing, the number of nodes/concepts increase rapidly — the time for accessing the Internet growing as the depth increased. In this section, we will evaluate the time for building concept map and the coverage of the Computer_security concept map generated in this research.

4.1.1 Concept map building time

Tab. 4.1 shows the total time (execution time and the Internet accessing time) for concept building in various depth. As depth growing, the number of nodes/concepts increase rapidly — the time for execution remained on a slow-growth while the time for accessing the Internet growing rapidly. The average execution time for the seven concept maps with different depth is 1.432918 seconds. The average time for accessing the Internet of the seven concept maps with different depth is 329.938169 seconds. Especially, when on the relatively large depth, the time for accessing the Internet is almost 779 times slower than the execution time as we observed from Tab. 4.1, the main factor of the concept map building is the Internet accessing time.

Fig.4.1shows that, as depth growing, the total time for the Computer_security concept map building increase rapidly. And combine the reservation from Tab. 4.1, there is no doubt that, the main factor of the concept map building is the Internet accessing time.

We have constructed sub concept map with seven different keywords. The result showed in Tab.4.2 that as depth growing, the total time for the sub

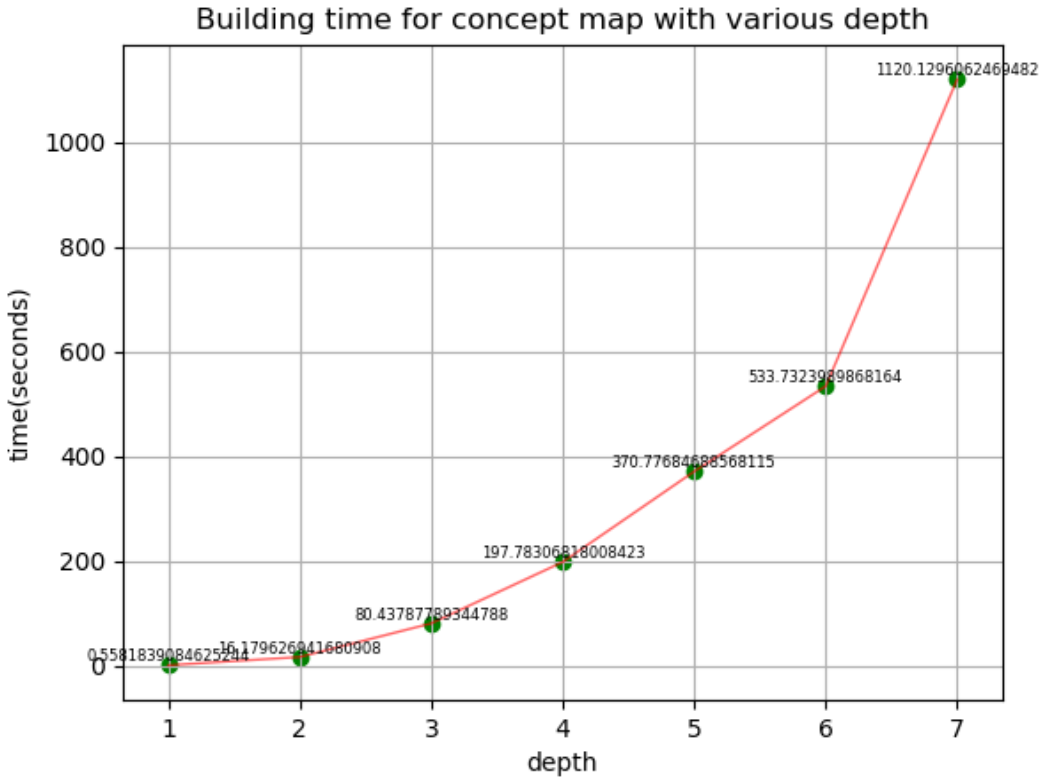

Figure 4.1: Building time for Computer_security concept with various depth

Table 4.1: Concept map building time

depth	concepts number	execution time(s)	Internet accessing time(s)
1	23	0.01110610	0.54707691
2	126	0.07438310	16.10524294
3	331	0.35431300	80.08356489
4	618	0.97561000	196.80745818
5	884	1.83946000	368.93737889
6	1347	2.63489700	531.09750199
7	2640	4.14064700	1115.98895925

concept map building remained stable. The average time of sub concept building is 0.51459295 seconds.

And combine the reservation from Tab. 4.1, there is no doubt that, the main factor of the concept map building is the Internet accessing time. It makes sense that the strategy of construct sub concept map from Comper_security map is meaningful and useful.

Table 4.2: Sub concept map building time

concept	building time(s)	depth
Authentication	0.42738413	1
Computer_viruses	0.45832491	2
Malware_in_fiction	0.47378087	3
Malware	0.56676579	4
Authentication_methods	0.47324681	5
Authentication	0.42738414	6
Computer_security	0.77526402	7

Table 4.3: Time comparison of concept map building between from the Internet and the local data

concept	accessing from Internet (s)	extracted from local(s)
Malware	37.76149082	0.37177896
Computer_security	1115.98895925	0.77526402

Tab.4.3 showed the time for building concept map from the Internet and the local data (seven depth Computer_security concept map) of two concepts: Malware and Computer _security. It is evident that the strategy of construct sub concept map from Comper_security map could improve the user experience since waiting time reduced.

4.2 Concept map coverage

We evaluated building time for the Computer_security concept map in previous, and in this section, we will assess the coverage of this concept map by comparing it with ESET Cybersecurity Awareness Training project and CompTIA Security+ Study Guide.

ESET Cybersecurity Awareness Training mentioned in Chapter 2 advertises itself that it teaches everything the employees need to understand to help make your company's cybersecurity safe. ESET provides training from 5 aspects: threats overview, password safety, internet protection, email protection, and preventive measures. In this thesis, we extracted keywords from ESET Cybersecurity Awareness Training and compared to the concept map generated in this thesis.

Tab. 4.4 shows the matching results with ESET Cybersecurity Awareness Training. We found the matchings of each five aspects in Computer_security map built in Chapter 3. One advantage of the built Computer_security concept map is that it has more detailed concepts. For example, the Malware in ESET only contains itself, while from the built concept map, we can get much more concepts, referred in Tab. 3.5.

CompTIA Security+ is a necessary security certification for IT professionals. CompTIA Security+ study guide is a book prepared for security technologies who want earn the Security+ certification[22]. This book has 12 Chapters, provides the knowledge base, and skills range from physical security and software security.

Tab. 4.5 shows the matching results with professional IT book, CompTIA Security+ study guide. We found the most matches in our concept map expect for the *Securing the Cloud* and *Security Administraction*. Analysis why there are no matches of *Securing the Cloud*. There is a concept *Cloud_computing_security* existed in DBpedia(in Articles), but our built concept map does not contain it. We constructed the hierarchy concept map from the Categories of the DBpedia. But there is no *Cloud_computing_security* in Categories. The linked concepts of *Cloud_computing_security* in Categories are Computer_security and Cloud_computing. In Chapter 3, when we link the Computer_security from Categories to Articles, even it has multi-

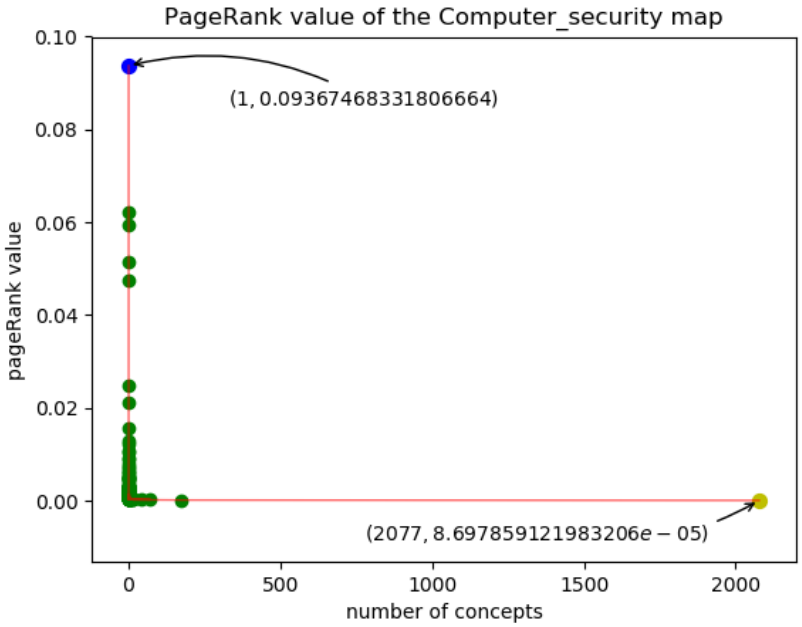

Figure 4.2: PageRank distribution of the Computer_security concept map

ple linked concepts, we only choose the most approximate concept: Computer_security in Articles (filtered out the *Cloud_computing_security*). For another linked concept to *Cloud_computing_security*, the Cloud_computing which is not the descendant concept of Computer_security, but it's a kind of top concept. So it is not in the built concept map.

4.3 PageRank evaluation

Fig. 4.2 shows the distribution of concepts's PageRank value in Computer_security map ($depth = 7$). As we can observe from it, the amount of concepts with small PageRank value is big, while the amount of concepts with high PageRank value is small.The blue point in Fig. 4.2 means that only one concept has PageRank of 0.09367468331806664, which is the Computer_security concept. The yellow point means that more than two thousand concepts share the same PageRank of 8.697859121983206-05, which is the smallest PageRank value in this concept map. Fig. 4.2 corresponds to Tab. 3.1 in Chapter 3, as depth growing, the nodes in each level increased fast. The concepts on the top level have higher PageRank value than the leaf concepts or the concept on the low level.

Table 4.4: Concept map coverage vs. ESET Training topics

ESET concepts		**concept map concepts**	**matching**
threats overview	malware	malware	◯
	viruses	compute_viruses ,antivirus_software	◯
	worms	compute_worms,email_worms	◯
	trojans	AIDS_(Trojan_horse)	◯
	ransomware	ransomware	◯
	rootkit	rootkit	◯
	spyware	spyware	◯
	social engineering	engineering_failures,reliability_engineering,software_engineering_costs...	◯
password safety	password	password_authentication,password_managers,password_cracking_software	◯
	access	access_control	◯
	authentication	authentication	◯
Internet protection		Internet_security,Internet_privacy_software	◯
email protection		email_worms, email_authentication,email_hacking	◯
preventive measures	spam filter	spam_filtering	◯
	password manager	password_managers	◯

Table 4.5: Concept map coverage vs. CompTIA Security+

CompTIA Security+	**concept map concepts**	**matching**
Chapter1: Managing risk	IT risk management	◯
Chapter2: Monitoring and Diagnosing Networks	computer network security, network analyzers, virtual private networks	◯
Chapter3: Understanding Devices and Infrastructure	firewall software,ripple gateways	◯
Chapter4: Identity and Access Management	identity management, access control	◯
Chapter5: Wireless Network Threats	rogue software	◯
Chapter6: Securing the Cloud		—
Chapter 7: Host, Data, and Application Security	data security	◯
Chapter8: Cryptography	cryptography	◯
Chapter9: Threats, Attacks, and Vulnerabilities	cyberattacks, cryptographic attackst,computer viruses, malware, computer worms, rootkit, adware, spyware,DOS viruses,domain hacks	◯
Chapter10: Social Engineering and Other Foes	engineering failures,reliability engineeringaccess control	◯
Chapter11: Security Administration		—
Chapter12: Disaster Recovery and Incident Response	back up,intrusion detection systemsspam filtert,port scanners,identity management	◯

Table 4.6: System feature evaluation: tradition adaptive learning system vs. system prototype

tradition adaptive system	**adaptive system prototype**
1. Expert model	✓✓training materials: concept map, question set in Section Chapter 3 timely updated, extended
2. Learner model	✓kinds of information of the learners: learner model in Section 3.6 only contains the understanding of the questions
3. Instructive model	✓✓combines the previous two models to show the next content update in Section 3.6
4. Instructional environment	✓user interface or training platform command line interface semi-interacted with Moodle

4.4 Adaptive training system evaluation

Chapter 2 introduced that tradition adaptive learning system usually divided into Expert model, Learner model, Instructive model, and Instructional environment. Tab.4.6 combined the tradition adaptive learning system with the adaptive system prototype in this research.

The adaptive system prototype in this research implemented all the primary function of the four modules. For the expert model, this prototype can dynamically and timely update the training materials and can extend the training coverage. For the learner model, traditional one often contains other information, such as age, educational background and so on. In this prototype, the learner model only includes one kind of information, but it can extend in the future. For instructive model, this prototype combines the processed concept in Chapter 3 and feedbacks from learner to provide the next question. As the survey did in research background, it is innovative to conduct adaptive training in this way. For the instructional environment, the practical project usually has a fancy interface. For the instructional environment, the practical project usually has a fancy interface. This research prototype provided the learner with a command line interface for full interact.

This section conducted *Malware* adaptive training to check whether this system can fulfill the purpose of improving cybersecurity people's awareness.This training consisted of 6 training concepts (Table 3.6) related to *Malware*. The training finished when the learner understood all the related concepts. And 8 learners (4 learners with cybersecurity background while the rest of the learners without cybersecurity background) participated in

this user experience survey.

The number of quizzes in Fig. 4.3 revealed how many small quizzes that the learner took to handle the malware concept. The back line Fig. 4.3 is the minimum times learner to handle the Malware training. The average quiz number of the cybersecurity background learner is 2.25, and the average quiz number of the none cybersecurity background learner is 3.5. There is a gap between cybersecurity background and none cybersecurity background. Even the text processing is not so advanced; the learner can understand the questions and made the right choices after training. Fig. 4.4 illustrated the Fig. 4.3, and it explains how many questions the learner answered wrong.

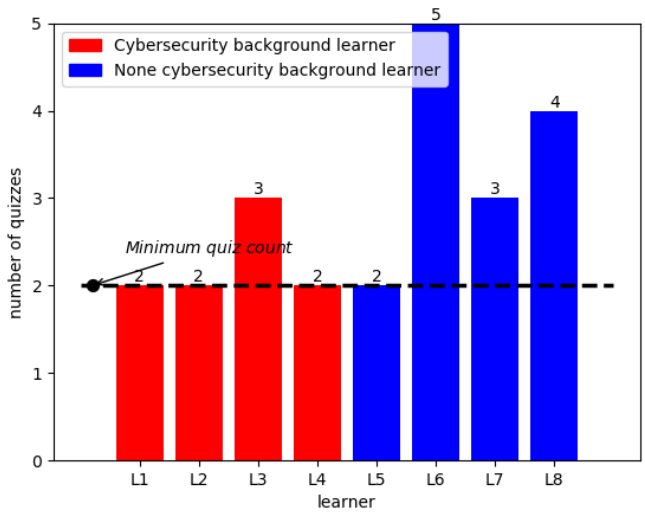

Figure 4.3: The quiz count for each learner to take the Malware training

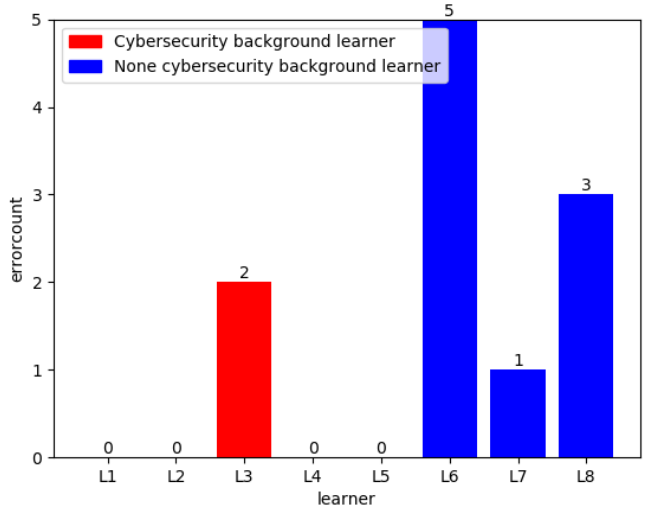

Figure 4.4: Errorcount

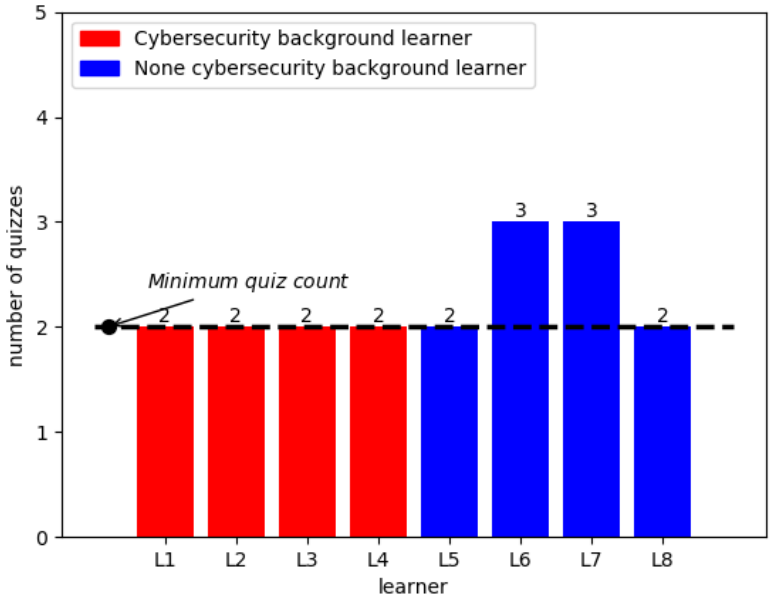

Figure 4.5: Retest: The quiz count for each learner to take the Malware training

Hermann Ebbinghaus hypothesized the famous forgetting curve in 1885. The curve demonstrated how the memory of data declines over time when there is no attempt to reinforce it [23]. Further, assessing the training results, this thesis retest eight learners in Fig. 4.3 21 days later. The retest results show improvement of the learners. All cybersecurity background learner finished the Malware training in minimum quiz count. For the noncybersecurity background learner, the quiz count declined. Especially the quiz count of the learner 6 decreased from 5 to 3. Generally speaking, the adaptive training system prototype proposed in this research helped learners to improve their cybersecurity awareness.

Chapter 5

Conclusion and Future Work

In common sense, cybersecurity which we often talked about is focused on the technologies to combat cybersecurity attacks or threats, especially hardware and software — the considerable amount of money spent on cybersecurity to protect people, company and organization from cyber attacks. Expensive and sophisticated systems won't play much good without the human factor. Human is the primary vulnerability in cybersecurity. We can not ignore the importance of the human factor.

This research started to construct Computer_security concept map from the LOD database DBpedia, dynamically and timely updated. DBpedia has a wide topic coverage, and it is possible to get much further information since it interlinked with other kinds of open datasets. After elevated in Chapter 4, along with depth growing, the number of nodes/concepts increase rapidly and the time for accessing the Internet growing immediately, especially, when on the relatively large depth. Timely update the big concept map make the cybersecurity materials is fresh. Constructed sub concept map from the big one save much time in the training.

In this research, we could timely updated cybersecurity materials form the LOD datasets. In Chapter 3, we proposed the way that how to build the big concept map and store this big concept map in local, constructed sub concept map from the big concept map based on the learner need. As evaluation made in Chapter 4, along with depth growing, the number of nodes/concepts increase rapidly and the time for accessing the Internet growing immediately, especially, when on the relatively large depth, the time for accessing the Internet is almost 779 times slower than the execution time. Timely update the big concept map make the cybersecurity materials is fresh. Constructed sub concept map from the big one save much time in the training.

We employed the Page Rank Algorithm in Chapter 3 to calculate the importance of each concept node on the concept map to conduct adaptive

awareness training later. The data nodes that have higher importance have higher priority for cybersecurity awareness training.

Filtered the irrelevant concepts and no definition concepts. This research constructed the hierarchy concept map from the Categories and the definition from the Articles in DBpedia. Before querying the definition from the Articles, first links Categories with Articles.

We proposed a simple way to do adaptive training and implemented an adaptive awareness training system prototype. The level and the PageRank value of the processed concept map, combined with the learner knowledge provided the main idea of the adaptive awareness training. Question creation and text processing made training system into actual practice. Learner model used to track the understandings to correct knowledge.

The future work of this research includes the following several tasks:

a. Get much further information in the future. As this research mentioned before, DBpedia interlink with other kinds of open datasets. In the future, more cybersecurity-related datasets hope to can be accessed.

b. SPARQL has limitations as a simple query language, compared to the programming language. Some interesting extended SPARQL research had been done on missing features, such as recursion. The extended SPARQL may be used in the future to reduce the Internet accessing time of the system[20].

c. Natural language processing must be used in text processing to generate questions.

d. Improve Learner model in the future. The learner model can be extended to contain kinds of information of the learners, such as domain knowledge, learning performance, interests, preference, goal, tasks, background and so on. And use the advanced learner model to improve practice training.

e. Item response theory (IRT) can used to improve the adaptive training. Estimate a learner's ability and determine the most relevant questions used to train a learner. From IRT, learner's response is a stochastic process in a test or an assessment which is the probability of answering correct or not, depends on kinds factors, such as some cognitive, emotion status, previous performance and so on. IRT is considered as superior of classical test theory, which based on the application of related mathematical models in data testing[21]. As started, the Rasch model can be the first try to difficulty and ability in this thesis.

f. Interact with Moodle. Add training content to Moodle by using CyLMS and get feedback from Moodle for update.

For more information, this research is open-sourced on GitHub.

Bibliography

[1] Jeremy Schwartz. *Infographic: 2016 State of Privacy and Security Awareness*, https://www.mediapro.com/blog/infographic-2016-privacy-security-awareness-iq/,(2016)

[2] FraudWatch International. *What is Cyber Security Awareness Training and Why is it so Important?*, https://fraudwatchinternational.com/security-awareness/what-is-cyber-security-awareness-training/,(2018)

[3] Wikipedia contributors. *Simple Knowledge Organization System— Wikipedia, The Free Encyclopedia*,`https://en.wikipedia.org/w/index.php?title=Simple_Knowledge_Organization_System&oldid=864344004/`,(2018)

[4] Wikipedia contributors. *Web 2.0— Wikipedia, The Free Encyclopedia*, `https://en.wikipedia.org/wiki/Web_2.0`,()

[5] Christian Bizer, Tom Heath and Tim Berners Lee. *International Journal on Semantic Web and Information Systems*, vol.5,

[6] Wikipedia contributors. *Semantic Web Wikipedia, The Free Encyclopedia*,`https://en.wikipedia.org/wiki/Semantic_Web`

[7] Nupur Choudhury. *World Wide Web and Its Journey from Web 1.0 to Web 4.0*, ,()

[8] Wikipedia contributors. *DBpedia Wikipedia, The Free Encyclopedia*,`https://en.wikipedia.org/w/index.php?title=DBpedia&oldid=845019184`,(2018)

[9] Auer, Sören and Bizer, Christian and Kobilarov, Georgi and Lehmann, Jens and Cyganiak, Richard and Ives, Zachary. *DBpedia: A Nucleus for a Web of Open Data*,Lecture Notes in Computer Science, (2007)

[10] W3C, RDF working grou. *RDF Semantics, W3C Recommendation 10 February 2004*,https://www.w3.org/2001/sw/RDFCore/TR/WD-rdf-mt-20030117/,(2004)

[11] W3C, RDF working grou. *RDF*,https://www.w3.org/RDF/,()

[12] W3C, RDF working grou. *RDF 1.1 Concepts and Abstract Syntax*,https://www.w3.org/TR/rdf11-concepts/,(2014)

[13] Wikipedia contributors. *RDF Schema Wikipedia, The Free Encyclopedia*,`https://en.wikipedia.org/w/index.php?title=RDF_Schema&oldid=863683076`,(2018)

[14] O'Reilly Media. *Learning SPARQL: Querying and Updating with SPARQL1.1*,(2013)

[15] W3C. *SPARQL 1.1 Query Language*,https://www.w3.org/TR/sparql11-query/,(2013)

[16] Brin, Sergey and Page, Lawrence.The Anatomy of a Large-scale Hypertextual Web Search Engine, *JComput. Netw. ISDN Syst.*, vol.30, no.yy,

[17] Wikipedia contributors. *Adaptive learning Wikipedia, The Free Encyclopedia*,`https://en.wikipedia.org/w/index.php?title=Adaptive_learning&oldid=840739618`,(2018)

[18] Alistair Miles and Sean Bechhofer. *SKOS Simple Knowledge Organization System Namespace Document - HTML Variant*,https://www.w3.org/2009/08/skos-reference/skos.html,(2009)

[19] Wikipedia contributorsr. *Simple Knowledge Organization System*,`https://en.wikipedia.org/wiki/Simple_Knowledge_Organization_System`,(2009)

[20] Maurizio Atzori. *Computing Recursive SPARQL Queries*,(2014)

[21] Wikipedia contributors. *Item response theory*,`https://en.wikipedia.org/wiki/Item_response_theory`,(2000)

[22] Emmett Dulaney and Chuck Easttom. *CompTIA Security+ Study Guide: Exam SY0-501 7th Edition*,Sybex and Jim Minatelr, (2018)

[23] Wikipedia contributors. *Hermann Ebbinghaus Wikipedia, The Free Encyclopedia*,`https://en.wikipedia.org/wiki/Hermann_Ebbinghaus`,(2019)