Personalized Learning in a Virtual Hands-on Lab Platform for Computer Science Education

by

Yuli Deng

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved November 2021 by the
Graduate Supervisory Committee:

Dijiang Huang, Chair
Baoxin Li
Ming Zhao
Sharon Hsiao

ARIZONA STATE UNIVERSITY

December 2021

ABSTRACT

Personalized learning is gaining popularity in online computer science education due to its characteristics of pacing the learning progress and adapting the instructional approach to each individual learner from a diverse background. Among various instructional methods in computer science education, hands-on labs have unique requirements of understanding learners' behavior and assessing learners' performance for personalization. Hands-on labs are a critical learning approach for cybersecurity education. It provides real-world complex problem scenarios and helps learners develop a deeper understanding of knowledge and concepts while solving real-world problems. But there are unique challenges when using hands-on labs for cybersecurity education. Existing hands-on lab exercises materials are usually managed in a problem-centric fashion, while it lacks a coherent way to manage existing labs and provide productive lab exercising plans for cybersecurity learners. To solve these challenges, a personalized learning platform called ThoTh Lab specifically designed for computer science hands-on labs in a cloud environment is established. ThoTh Lab can identify the learning style from student activities and adapt learning material accordingly. With the awareness of student learning styles, instructors are able to use techniques more suitable for the specific student, and hence, improve the speed and quality of the learning process. ThoTh Lab also provides student performance prediction, which allows the instructors to change the learning progress and take other measurements to help the students timely. A knowledge graph in the cybersecurity domain is also constructed using Natural language processing (NLP) technologies including word embedding and hyperlink-based concept mining. This knowledge graph is then utilized during the regular learning process to build a personalized

lab recommendation system by suggesting relevant labs based on students' past learning history to maximize their learning outcomes. To evaluate ThoTh Lab, several in-class experiments were carried out in cybersecurity classes for both graduate and undergraduate students at Arizona State University and data was collected over several semesters. The case studies show that, by leveraging the personalized lab platform, students tend to be more absorbed in a lab project, show more interest in the cybersecurity area, spend more effort on the project and gain enhanced learning outcomes.

ACKNOWLEDGMENTS

My deepest and foremost sincere gratitude goes to my advisor Dr. Dijiang Huang for his constant guidance and encouragement. He walked me through all the stages of the writing of this dissertation, and provided an incredibly collaborative, supportive, and friendly environment throughout my Ph.D. study. Without Dr. Huang's consistent and illuminating guidance, my knowledge and skills as a researcher could not have reached their present form.

I would also like to express my heartfelt grateful to my committee members, Dr. Baoxin Li, Dr. Ming Zhao, Dr. Sharon Hsiao for their invaluable comments, and critical introspection of my work. They instructed and helped me a lot in the past years.

The insightful inputs I have received from my both collaborators and colleagues in the Secure Networking and Computing (SNAC) lab at ASU are also very much appreciated. Thus, I want to thank my previous and current collaborators and colleagues, Dr. Chun-Jen (James) Chung, Dr. Duo Lu, Fanjie Lin, Zhen Zeng, Dr. Ankur Chowdhary, Abdulhakim Sabur, Kritshekhar Jha, Garima Agrawal, Sowmya Myneni, and Neha Vadnere. None of this would have been possible without the help of each of them.

Lastly, my especial thanks would go to my beloved family, my source of inspiration, for their loving considerations and confidence in me throughout the journey. My family has been extremely supportive of me and has made countless sacrifices to help me get to this point.

TABLE OF CONTENTS

Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Overview

The age of “one-size-fits-all” has passed, and personalization becomes the new norm in almost every aspect of our daily life, including learning. Personalized learning requires the instructor to pay attention to individual characteristics, such as learning style, learning preference, skill level, and previous knowledge, so as to adopt different learning materials and instructing techniques to each student. With the advent of personal computer and the Internet, many personalized e-learning systems were developed to accommodate the diverse needs of the students in various fields. Among these systems, the majority are based on traditional Learning Management Systems (LMS), whose functionalities are centered around the learning program management, learning content delivery, and student performance assessment based on written assignments as well as exams. A few systems address the needs of collaborative learning by providing online communication features like discussion board. However, most of them do not effectively support hands-on laboratory, which plays an important role in project-oriented education in STEM areas, especially, in cybersecurity education.

Starting from the very early stage of computer science education, hands-on labs have shown its significance in training problem-solving skills due to its unique feature of asking the student to apply the acquired knowledge actively. This is especially true in cybersecurity education, wherein such hands-on laboratories the students can put the learned concepts in the classroom into practice, observe the cause and consequences of system breaches, learn from challenges, and improve skills based on their own mistakes.

In the last decade, with the rapid development of cloud virtualization, Internet, and human-computer interaction technologies, more and more hands-on labs for computer science and cybersecurity education have been designed to fit the cloud environment and made available online. Platforms such as Amazon AWS Education [1], MyIT Lab by Pearson [2], Microsoft Azure in Education [3] and Cloud Lab [4] support the student to remotely access computing resources for a hands-on lab in programming, networking, and cybersecurity. However, they are merely cloud resources used for an educational purpose which only transform physical labs into an online virtual form, not dedicated personalized learning solutions that can manage and adapt learning process to the student.

Another part of challenge of virtual hands-on laboratories is lab material management and organization. Existing lab materials are mainly managed in a problem-centric fashion, in which instructors arrange learning and corresponding lab materials based on a specific topic in security area. However, the inter-lab dependencies are usually complicated and unclear, which hinders both students and instructors to manage learning and teaching materials in a coherent way. It is challenging to build an effective and adaptive learning schedule for students according to their personal background and learning targets: First, efficient cybersecurity education heavily relies on hands-on labs since it focuses more on practical problem-solving skills instead of theory and models. In addition, it is more difficult to organize lab materials than textbooks, let alone manage a complicated experiment environment with multiple hosts, switches, routers, and cables. Second, due to inherent diversities in knowledge and skill sets in cybersecurity education, it is difficult to personalize the learning process and keep track of individual student's learning progress. Third, for instructors, the knowledge sets and instructing materials must be kept up to date

to cope with the emerging new vulnerabilities, attacks, and defense solutions. As a result, it is a continuing process to provide improved learning guidance and plan for students to keep up with the evolving of cybersecurity technologies.

I developed ThoTh Lab, a personalized learning framework for cybersecurity hands-on labs in cloud environment. It is developed in Arizona State University for upper-level computer science courses. To summarize, in ThoTh Lab:

- It provides a wide range of applications, virtual machines, and network devices that can fully simulate a real-world hands-on lab in cybersecurity education, where the students can set up experiments, build solutions and retrieve experiment results.
- A pure web front-end for ThoTh Lab was constructed to allow flexible access anywhere, anytime for students and instructor to manage lab efficiently through the Internet.
- A learning behavior analyzer, a learning style classifier, and an adaptive learning content manager are built and utilized machine learning models and the web page interaction data as well as virtual machine logs to understand student activity, identify the learning style, and adjust learning materials accordingly.
- A student learning performance assessment and prediction module was constructed for instructors to better understand the student learning progress and adapt accordingly, so as to improve students' learning experience and efficiency.
- A knowledge graph of concepts and terminologies of cybersecurity was constructed based on large amount of public cybersecurity contents, such as Wikipedia and publicly available cybersecurity lab descriptions. Nodes of the knowledge graph and their dependency relationship are obtained by mining the public cybersecurity contents and

security concepts from many cybersecurity glossaries fine-tuned with reading materials and hands-on lab instructions used in the offered security courses.

- A lab recommendation system was built to make real-time suggestion for the hands-on lab environment. This system can make recommendations by exploiting the similarity relationship between nodes in the knowledge graph and the association between various knowledge graph nodes and lab instructions.
- A Q&A Module was also constructed to answer student questions and query during lab session, to provide students with background knowledge and tutorial materials related to lab tasks in real-time.

ThoTh Lab is the first of its kind, which provides a complete personalized learning solution of learning style identification, learning performance assessment, adaptive learning content recommendation, and virtual hands-on laboratory.

The ultimate goal of teaching is to promote learning. This is also the prime objective of ThoTh Lab. First, ThoTh Lab is a resource and service sharing platform that provides segregated virtual systems for students to do hands-on labs their own networking environments. In addition to this, it also provided a user-friendly, easy-to-use graphical user interface that will attract more students to do hands-on labs remotely and increase students' awareness of computer network and security. Last and most importantly, it introduced personalized learning to a Computer Science Lab platform. Our ultimate goal is to improve students' learning experience during the lab sessions, motivate them to put more effort into hands-on labs, and improve their final learning outcome.

A list of experimental studies was carried in class lab sessions to test and evaluated the ThoTh Lab system. The experimental results show that given a reasonable amount of

student data, the proposed framework can identify learners' individual characteristics and provide an accurate assessment. The case studies result also shows the educational benefits in terms of enhanced learning performance, a higher level of student participation and increased satisfaction with the ThoTh Lab personalized learning framework.

1.2 Organization of Dissertation

The dissertation is organized as follows: background and the current state of the art related to this dissertation research will be presented in Chapter 2. A detail illustration of the approach will be presented in Chapter 3. The experiments results will be presented in Chapter 4. The conclusion of this dissertation work and future research directions will be presented in Chapter 5.

CHAPTER 2

BACKGROUND & STATE OF ART

This chapter covered background information about cloud computing, personalized learning, learning style, student performance prediction, natural language processing (NLP) machine learning techniques, blockchain and discussed start of art solutions available.

2.1 Cloud Computing

Cloud Computing is an approach aims at delivering compute, network, and storage infrastructure resources, services, platforms, and applications to users over network. These infrastructure resources, services, and applications are sourced from clouds, which are pools of virtual resources orchestrated by management and automation software so they can be accessed by users on-demand supported by automatic scaling and dynamic resource allocation. Virtualization is technology that enables cloud, it allows user to create multiple simulated environments or dedicated resources from a single, physical hardware system and provide them as resource in cloud.

When applying cloud computing in my search, I fully utilize the virtualization capacities of cloud to provide dedicated and contained experimental environment with multiple VMs and multiple virtual networks for each learner. The system offers a Web-based management portal for instructors and students to manage and create virtual resources in a user-friendly fashion. The virtual resources can be reconfigured throughout the course to

introduce new experiments. The system built provides an interactive Web GUI for network constructions and reconfigurations and experiments deployments.

2.1.1 Cloud Computing in CS Education

In education, cloud computing caters for desirable properties to provide e-learning services, especially in scenarios where these services are computer-intensive (virtual worlds, simulations, video streaming, etc.), or are offered in a high-scale way, as in Massive Open Online Courses (MOOCs). The cloud can provide students and teachers with tools to deploy computing resources on-demand for lectures and labs according to their learning needs. For instance, teachers can create virtual machines on demand with pre-installed software to deploy computing laboratories rapidly [41]. Some educational institutions are already using cloud computing to offer collaboration tools and data storage for students and to host institutional Virtual Learning Environments (VLEs) [42]. Other affordances of cloud computing may yield new learning scenarios where ubiquity, advanced online tools and collaboration come together to create innovative opportunities for education. On the other hand, cloud computing brings new risks when compared to the conventional IT model such as security, performance, or interoperability that now must be considered. The adoption of cloud computing in education has come hand in hand with an important research effort. There are a great number of scientific contributions that address the topic and challenges from different perspectives trying to harness cloud computing services for education. These challenges can be either technical issues (i.e., how to improve the cloud

technology itself to meet domain-specific needs) or domain-specific opportunities (i.e., how to leverage cloud computing services for pedagogical uses).

2.2 State of Art of Virtual Laboratories

Since virtual laboratories have numerous advantages over traditional and remote labs, virtual laboratory design and development became a widely researched topic in the past few years. Many research groups, universities and even members of the industry developed their own virtual laboratories, covering a very wide area of disciplines, such as biology [43], [44], physical sciences [45], [46], engineering [51], [52], [53], [54] and computer sciences [47], [48], [49], [50]. Since the initial introductions of the virtual laboratory concepts, research on the effectiveness of virtual laboratories confirmed that properly designed virtual simulation tools can enhance students' learning processes [55][56]. As a result, virtual laboratories have been implemented and/or studied across the wide variety of disciplines. However, one needs to be careful when using the term "virtual lab" since there is no set definition for "virtual lab." For instance, in some papers, the term "virtual labs" refers to the remote operation of physical labs [57]. In current scientific literature, two major categories of virtual laboratories can be distinguished. In the first category [58] [59], there is virtual laboratory implementations that were designed and suitable for a single specific purpose, i.e., for a single experiment. These implementations do not employ systematic approaches on design, e.g., does not take into account the re usability or different components. Instead, the focus is on the experiment itself and the software components that are already playing a role in this experiment is also used to implement the

necessary services for the remote laboratory, e.g., the webserver. Most of the time, this approach yields quick results because the teacher or researcher are already familiar with the software, but in the long run it is a source of multiple problems. First, the software, which is perfectly suitable for the experiment, e.g., LabVIEW or MATLAB may contain a module to implement a webserver. However, these were not designed to handle the challenges, e.g., a large number of concurrent requests that webserver software was designed for. Second, it is impossible to integrate these components with external services, like authentication and authorization, accounting, and data storage. Third, it is very hard to create a consistent user interface for the end-users of the system. In the second category [60] [61], there are proposed virtual laboratory system designs, some of them with reference implementations, which were created by following a systematic design approach, employing design patterns and best practices from the field of web-development. This approach successfully eliminates the problems mentioned in the first category. However, it poses a new requirement against the teams who want to implement a new experiment, namely that they have to learn the usage of new tools that are specific to web development. In computer science disciplines, this is often not a problem because teachers already have some knowledge and expertise in this area. However, in other disciplines, e.g., astrology, chemistry and medicine, where virtual laboratories are also very useful, this problem makes it hard to involve teachers in creating new virtual experiments. Recently it is a subject of active research to fill the gap between these two categories. The research challenges are how to cover the gap between the two categories and merge the effort from both side of researchers.

2.3 Personalized Learning

Personalized learning is increasingly recognized as a promising strategy to close achievement gaps, increase student engagement, and prepare students as they become self-directed, lifelong learners by meeting their individual needs. While no single definition of personalized learning exists, the many definitions from leading experts share common general principles that include student voice and choice, customization to each student's strengths and needs, student agency, and flexibility of instruction. According to US Department of Education, Personalized learning refers to instruction in which the pace of learning and the instructional approach are optimized for the needs of each learner. Learning objectives, instructional approaches, and instructional content (and its sequencing) may all vary based on learner needs. In addition, learning activities are made available that are meaningful and relevant to learners, driven by their interests and often self-initiated [5].

2.4 Learning Style

As defined by Honey and Mumford [6], learning styles are "a description of the attitudes and behaviors" which determine an individual's preferred way of learning. Modeling and identifying learning styles is usually considered as the start point of personalized learning. In the past, several studies have proposed different ways to model learning styles, such as the model proposed by Honey and Mumford [6], Felder and Silverman [7]. Among these models, the system adopted the Felder-Silverman Learning Style Model (FSLSM) for engineering education due to its popularity and wide reception in e-learning environment

research. The FSLSM classifies students according to their position in several scales that evaluate how they perceive, process, and understand learning contents. The learning styles are defined in four dimensions, and each of them is represented as a pair of distinct learning styles. The first dimension considers through which sensory channel used by the learner to perceive external information most effectively -- either visual or verbal. The second dimension focuses on the learner's preferred method of processing information, either active or reflective. The third dimension considers how the learner progresses toward understanding -- either sequentially or globally. The fourth dimension defines what type of information that the learner preferentially perceives -- either sensory or intuitive. To identify the learning styles of a student, Felder and Silverman designed a questionnaire called “Index of Learning Styles (ILS)” to assess preferences of the student in all four dimensions. The questionnaire consists of 44 multiple choices questions, 11 questions for each dimension. FSLSM is one of the most frequently cited learning style models in the research area of computer-based and network-based education systems. There are also a few learning systems that are capable of adapting learning contents according to students' learning styles. But most of the approaches use the ILS as an online questionnaire to evaluate learning preferences as the first step, then present appropriate learning materials based on the answers of students. Answering the 44 questions is a time-consuming task, and consequently, it hurts user experience when applied directly in existing online personalized learning systems. Although the ILS questionnaire can tell us the general inclination of the learning styles of a student, the extent of the identified learning style cannot be estimated accurately because the student's answers are subjective.

With the advancement of educational data mining technologies, data-driven learning styles identification has been applied in several studies for personalizing learning. Liyanage et al. [8] presented a comparison of several data mining algorithms to detect student learning styles in a learning management system. [9] introduced a similar mechanism that uses k-nearest neighbor classification in an attempt to classify learning styles in a SCORM-compatible LMS environment. [10] used web usage mining and artificial neural network to identify students' learning styles in a web based LMS environment and created an adaptive user interface for it. [11] applied a simple rule-based student modelling approach to detect learning styles in a study with 127 students. [12] used feed-forward neural networks to detect learning styles. However, most of the research adopt FSLSM model with traditional online learning management systems, which does not provide noticeable support of remote hands-on labs for computer science education. The argue is that the difference in learning styles have a larger influence on the hands-on labs because in such situation the students are required to be actively engaged, self-motivated, and well-paced without supervision.

2.5 Problem-based Learning

PBL was pioneered by Barrows and originally used for medical education [62]. Over the years, the model has been adopted to teach concepts in other disciplines like architecture, law and business management [63][64][65]. PBL has also been identified as an efficient pedagogy for engineering education, where "engineering professions constantly deal with uncertainty, and with incomplete data and competing demands from clients, governments,

environmental groups, and the general public [66]." Studies show that PBL compensates learners with the engineering knowledge and skills they obtained despite the greater amount of work. Despite the greater amount of work, it was revealed that students learning with the PBL approach not only were benefited in the content area but also in generic skills such as leadership, analytical thinking, conflict management, and decision making [67]. Another study also shows that PBL is a promising approach in limited timed training, where a short time PBL training can be very effective when associating with technology projects [68].

2.6 State of Art of Student Performance Prediction

Student performance prediction is another important technology that facilitates personalized instructing for teachers. A few existing systems are able to predict student performance using data mining technologies based on student activity data and existing academic record. [13] successfully predicted student performance in a computer science course with data obtained from a learning management system and student profile. [14] employed linear regression to predict student's exam results base on source data of 103 variables collected during a classroom environment. [15] expand the scope of the data source by monitoring twenty different types of log data and combining several regression techniques in order to predict student grades in a remote education program. All studies described above are doing learning performance prediction by exploiting correlations between various features and the final scores of the students.

On the other hand, data mining techniques were also applied to develop performance prediction models. [16] demonstrates the capability to predicted passing or failing grade for a student in an online education program using neural network models based on log data captured by a MOOC platform [17]. Researchers also have performed a wide range of comparison of various machine learning models on predicting student learning performance. From an education practitioner's perspective, neural network and SVM models are black-box models, where the internal decision-making procedure are not interpret-able, and they are not easy to implement correctly. Thus, instructors cannot gain much useful information other than the prediction result of a score. Other researchers carried studies to explored domain knowledge to improve prediction by the model, such as rule-based predictor, belief network, logic programming and reasoning process. These white-box methods provide explanations for all the classification results. For example, Bayesian Networks have been used to predict students learning performance using log data [18].

ML is defined as the ability of a machine to vary the outcome of a situation or behavior based on knowledge or observation. Using algorithms that iteratively learn from data, ML enhances many technologies to analyze the data immediately as it is collected, to accurately identify previously known and never-before seen patterns.

2.7 Natural Language Processing

Natural Language Processing is a branch of artificial intelligence that deals with analyzing, understanding, and generating the languages that humans use naturally in order to interface with computers in both written and spoken contexts using natural human languages instead of computer languages.

For the computer to understand natural language and the knowledge and concepts within, there has to be a way to represent words. Traditionally, NLP systems treat words as discrete symbols which leads to data sparsity and usually means that one may need more data in order to successfully train statistical models. Word embedding is a set of language modeling techniques to represent words with a vector in a low dimensional space. Using vector representations makes natural language computer-readable, which enable us to perform powerful mathematical operations on words to detect their similarities.

In my research I mainly utilized a word embedding tool called Word2Vec [19]. Word2vec is a two-layer neural net that processes text. Its input is a text corpus, and its output are a set of vectors the purpose and usefulness of Word2vec is to group the vectors of similar words together in vector space. That is, it detects similarities mathematically. Word2vec creates vectors that are distributed numerical representations of word features, features such as the context of individual words. It does so without human intervention. Given enough data, usage, and contexts, Word2vec can make highly accurate guesses about a word's meaning based on past appearances. Those guesses can be used to establish a word's association with other words or cluster documents and classify them by topic. Those clusters can form the basis of search, sentiment analysis and recommendations. During my

research, I apply Word2Vec to process all the Lab content made available by us and other public sources to extract learning concepts and knowledge. I than create a knowledge graph composed with concepts extracted from lab and use the graph as a guidance for instructors and students. The mapping between the knowledge graph and labs hosted allow the lab platform to generate recommendations base on students' individual background and activity history.

2.7.1 Latent Semantic Analysis

The Latent Semantic Analysis (LSA) [69] is one of the most important bag-of-words methods. It describes each word in a vector space, where each word is represented based on its contextual usage to a document. LSA takes as input a training corpus formed by a collection of documents. A word by document co-occurrence matrix is constructed, which contains the distribution of occurrence of the different words and the documents. A mathematical transformation is usually applied to reduce the weight of uninformative high-frequency words in the words-documents matrix. Finally, a linear dimensionality reduction is implemented by a truncated Singular Value Decomposition (SVD) [70], which projects every word in a subspace with lower dimensions. The success of LSA in capturing the latent meaning of words comes from this low-dimensional mapping. LSA is widely used in the Natural Language Processing (NLP) domain.

2.8 Knowledge Graph

A knowledge graph is a programmatic way to model a knowledge domain with the help of subject-matter experts, data interlinking, and machine learning algorithms. Knowledge graph such as WordNet [20] is abundant graph model, its entity can be represented as a node and the link can be represented by the relationships between nodes.

Building a KG is a challenging task though efforts have been done in this area in recent years. There are two major approaches to develop the knowledge bases in education: the first approach primarily relies on individual professional expert, which involves manual work to a certain degree to determine the discrepancies among different professionals and then generate a corresponding consolidated graph. The first step is for each human expert to do text analysis and get a list of concepts, represented as labeled points, and a list of links between these nodes. By combining the lists of concepts and links, a small knowledge graph from a single author is generated, which is called an author graph. The next step is to combine graphs from various authors into one large graph by identifying common points with each other. When the texts of the nodes deal with the same subject, points with the same label are first identified. Then, human help is needed identity synonyms for the same concept and connect these synonyms together. There have been research efforts to describe and categorize knowledge and skills in cybersecurity area by a large board of professionals: Cybersecurity Curricular Guidelines [21], NIST NICE [22], NSA CAE Knowledge Units [23], etc. The outcome of these efforts are well-organized categories in tree structures, which provides clear guidance for human learners when exploring the area. However, it turns out to be significantly challenging for machine learning purposes as these structures

contain very limited semantic data that is readable to a machine. The other approach is to automate the generation process by gathering data from web pages and books which is achievable by computers rather without human interaction, e.g., Wikimind map [24]. There are various solutions been proposed in the last decade of research about building the KG: [25] have shown how to construct a knowledge base from Wikipedia in multiple languages; [26] gave a comprehensive review on training statistical models for large KG's, and further used them to predict new edges in the graph. Recently, attention has been drawn on word embedding for various learning tasks. While a word can be understood by a human being when it appears in the context, its numerical model has to be constructed based on the complex contexts using neural network. According to the previous work done by [27], two pages from Wikipedia are defined to be most similar when they have more common information being shared. As for other research, e.g., [28] showed that using the Anchor texts of Wikipedia led to better performance in learning the phrase vectors. [29] represented their work on constructing the specialized dictionary by using word2vec to train the Wikipedia data. Speer et al. [30] represented a knowledge graph - ConceptNet5.5, which combines several sources to acquire word embeddings by using distributional semantics, e.g., word2vec.

2.8.1 State of Art of Knowledge Graph as Cybersecurity Guidance

Many studies have been conducted on developing a cybersecurity curriculum or guide for universities: [71], [72], [73], [74]. Furthermore, a multitude of frameworks and learning objectives for cybersecurity have been established (e.g., CAE-CO [75], NICE

Cybersecurity Workforce Framework (NCWF) [76], ACM Joint Task Force on Cybersecurity Education [77]). Nevertheless, there is still a significant gap in maintaining and updating cybersecurity instruction guide at a practical level.

Frameworks such as NCWF and CAE-CO provide a detailed listing of knowledge and concepts required to succeed in a cybersecurity career. These sources of material are solid and are increasingly being recognized. However, adopts the baseline requirements or objectives of these frameworks makes learning mainly focus on science and literature topics instead of hands-on practical learning skills. Many institutes that offer cybersecurity programs still require a comprehensive guide to improve established learning guidelines. To meet these challenges, researchers adopted the knowledge graph as an AI tool to generate learning guides in an automotive fashion for students [78], [79]. Knowledge graph technology has drawn a lot of research attention in recent years [80]. Furthermore, information extraction and recommendation system are among the most popular real-world applications of the knowledge graph.

However, these approaches have their limitations. [79] requires significant human input during the knowledge graph construction stage to reduce errors, limiting feasibility in real-world applications for complex education areas. [78] uses embedding-based relation extraction approach to generate knowledge graph automatically from text data but suffers in accuracy and reliability due to its limited data source size as word embedding requires large-size text corpora to perform well.

2.9 Blockchain

Blockchain is one of the most promising technologies of the new. Blockchain is a type of distributed ledger in which value exchange transactions are sequentially grouped into blocks. Each block is chained to the previous one and immutably recorded across a peer-to-peer network, using cryptographic trust and assurance mechanisms. It maintains a coherent state, as agreed upon by all participants.

Public blockchains grant read access and ability to create transactions to all blockchain users. Users can transfer value without the expressed consent of the blockchain platform operator. The core property of these blockchains is censorship resistance, i.e., any valid transaction broadcast over a permission-less blockchain network would be included into the blockchain. Such blockchains are by their nature free for entry or exit both for users and application developers. The most prominent example of public blockchain is Bitcoin [38] – everyone is free to create a wallet, perform transactions with bitcoin units or become a miner (a nod, performing transaction verification functions for a fee in the form of newly created bitcoin units) by installing and using special publicly available software on its infrastructure.

Private blockchains limit access to the predefined list of known persons. Such persons should receive approval from a blockchain operator, thus the use of blockchain is restricted by end users and application developers. Such blockchains to a certain extent contradict to decentralized nature of blockchain technology itself, but still resemble certain advantages of this technology: transparency and resilience to attacks [39].

In this research, I utilized blockchain to create a blockchain-based digital content distribution system with copyright protection. Blockchain provides new paradigm for data storage security, based on the principle of decentralization. Its main features are:

- Transparency: all the data on blockchain is public, it cannot be arbitrarily tempered with and easily auditable.
- Redundancy: every user of the blockchain solution holds a copy of the data, thus it cannot be easily taken offline due to a system malfunction or malicious actions of third parties.
- Immutability: changing records in blockchain is prohibitively difficult and requires consensus provided in accordance with the protocol (e.g., by the majority of blockchain users). Thus, integrity of records is ensured by intrinsic properties of the underlying code rather than from the identities of system operators.

Some features of blockchain technologies: scarcity, trust, transparency, decentralized public records and smart contracts make such technology compatible with the fundamentals of copyright. Copyrights owners, like lab content authors in this case, can publish labs on blockchain creating an immutable record of initial ownership, and encode smart contracts to license the use of their own works. The concept is totally different from the conventional center operated rights management system. This means owner can control everything in the proposed system.

CHAPTER 3

SYSTEM DESIGN

The proposed ThoTh Lab framework, as shown in Figure 3.1, consists of a cloud-based virtual lab platform for cybersecurity education and a list of modules specially designed for different personalized learning functions that together achieve the contributions described in the Introduction. These modules are precisely explained in the rest of this chapter.

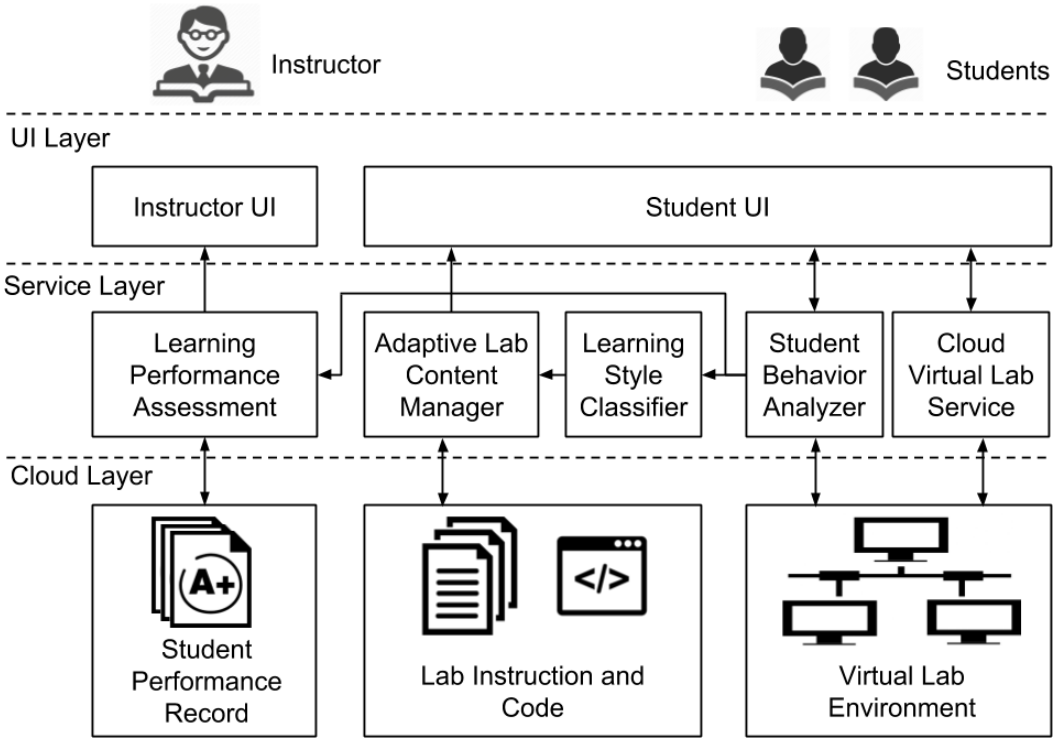


Figure 3.1: ThoTh Lab System Architecture.

The ThoTh Lab architecture contains three layers:

UI Layer: This layer presents two most important parts of a hands-on lab over the browser: (a) the virtual lab environment as virtual machines as well as virtual network, and (b) the

lab materials including instructions, code snippets, explanatory text, and figures. It is mainly developed in JavaScript, and it has two different views for instructors and students. The instructor can create a hands-on lab, by setting up the lab materials using a web-based editor and configuring a virtual lab environment by dragging and dropping virtual machines and virtual network devices into the canvas. Once the lab is created, it can be submitted to the back-end cloud virtual lab management services and allow students to enroll and practice. For the students, they can read the lab materials and access the virtual machines and network devices through the browser.

Service Layer: This layer glues the user interaction and back-end virtual machine, manages virtual resources, and provides services for certain functions for personalized learning. It is mainly implemented in PHP and Python. I leveraged my previous experience on microservice architecture and segment the system function into a few self-contained services. In this layer, the system (a) monitors user activities on the web UI and inside the virtual machine, (b) extracts high-level features from raw activity data, and (c) trigger learning style identification as well as lab content adaptation. More details are provided in later subsections.

Cloud Layer: This layer manages the cloud infrastructure, back-end services, user data, and lab materials. Besides, the system also hosts a repository of lab content with instructions and code in this layer using MongoDB, where the lab content can be flexibly adapted in different formats according to different learning styles. Around 60 labs are created and maintained by the system. Some of the lab contents are rearranged and edited from SEED Labs, and others are written by us from scratch covering emerging topics in the cybersecurity area, including recent research such as attacking gesture-based

authentication system, defending DDOS attack, deploying IDS in SDN environment. Moreover, the system also stores the student performance record in a secure database in this layer.

3.1 Cloud Virtual Lab Service

The cloud infrastructure of ThoTh Lab is built upon OpenStack, which is a widely used open-source cloud computing infrastructure platform. The back-end services contain various internal services for administration and management purposes. The virtual lab platform allows instructors and students to set up and access a lab environment of virtual machines and virtual network with maximal flexibility and easiness using remote and geographically distributed cloud resource, instead of physically set up a few computers and plug network cables into hosts, switches, and routers that are typically required by conducting a cybersecurity hands-on lab. Since such labs usually require multiple machines and special network topology for generating desired types of traffic, deploying a service, attack a server from a different machine, and etc., it is cumbersome and error-prone to configure physical devices, but extremely fast and cost-effective to set up with virtual resources in the cloud. The lab environment of each student is self-contained and can be accessed securely through an interactive web-based GUI. The student can sign-in to his or her virtual machines and network devices to change configurations and run any program in order to finish the tasks required by the lab instruction. The system keeps track of the logs and student activities over the web-based UI as well as inside the virtual machines for further analysis and personalized adaptation.

3.2 Student Behavior Analyzer

The student behavior analyzer is responsible for recording and understanding user behaviors based on low-level events such as simple features such as mouse click, mouse hover, command line activity and time spent inside a virtual machine, etc. It has three subcomponents. First, a JavaScript-based user behavior logger is implemented on the web page to monitor user's online activity. Second, a Logstash forwarder is installed inside each virtual machine of the student's lab environment to gather syslog, command line, and other activities. Third, the logged data is regularly analyzed a Python program to extract high-level student behavior features in a particular lab. Such features include session active time, lab requirement view time, and other features which has a potential correlation to the learning styles. Even though the same activity may repeat, the purpose of activity could be different depending on the lab context. Hence, it is necessary to collect and accumulate various activity patterns with the associated context for further learning performance assessment so as to examine the user's behaviors and deduce the patterns of users' meaningful behaviors.

3.3 Learning Style Classifier

The learning style classification module takes the output of user behavior analyzer (i.e., student behavior features), and use data mining models to identify different kind of users based on the FSLSM model discussed in section II. Before construction of the data mining

models, the system needs to select the features that are worth modeling and useful in classification. Though data mining methods like SVM classification do not require to understand the meaning of each feature, a few features are picked based on common sense. For example, to determine whether the student prefers reflective or active learning, the system analyze his/her participation in discussion systems and chatting service. For discussion forum, the system analyzes how often the student opens a new discussion, replies to other students' message, and reads the topics posted by other students. The system also collected general data which has an implicit correlation with learning styles that may help the learning style classification, like mouse clicking counts, keyboard inputs and syslog events in each virtual machine. The following features are currently used:

1 Mouse clicks count within Virtual Machine window.

2 Keyboard inputs count within Virtual Machine.

3 Virtual Machine syslog events' timestamps (when match with pre-defined event list) on the content navigation bar for each lab.

4 Hint bottom access counts during lab.

5 Quiz grades after each lab.

6 Group Chatting message counts during lab.

7 Discussion board topic access timestamps.

8 Discussion board new topic publishes count and replies to counts.

9 Virtual Machine bash history file (when match with pre-defined command list).

10 Timestamps when user access, exit lab content document, play videos, and click

After feature selection, a combination of SVM and Decision Tree models are then used for learning style classification. The system collects the data and label the learning style using

ILS Questionnaire [31] (APPENDIX A) to train the classifier. In total there are four different ways of categorizing the learning styles, and hence, ThoTh Lab has four independent sets of classifiers for these four identification tasks. It is well-known that ensemble of classifiers can improve the performance compared to using only individual constituent classifier. In particular, the system combines SVM and decision tree in the framework, as there is such a big difference in the fundamental model structure between SVM and decision tree. Also, both methods have shown good compatibility and performance in related applications. In the ensemble algorithm, the first step is to construct the constituent SVM classifier and the decision tree classifier from the training dataset. Then the testing data is classified by both algorithms independently. The final predicted label is derived from the output of each constituent classifiers. If both classifiers output the same label, the label will be kept as the result. Otherwise, the framework runs the following steps:

1 If one of the prediction models classified the testing sample as neutral, neutral will be keep the label of classification.

2 Find PSVM=n(ErrDT)/n(ASVM), where n(ErrDT) is the total number of training data, whose class label predicted by SVM is correct, and decision tree prediction is incorrect. n(ASVM) is the total number of training data whose class label predicted by SVM is correct.

3 Find PDT=n(ErrSVM)/n(ADT), where n(ErrSVM) is the total number of training data, whose class label predicted by decision tree is correct, but SVM prediction is incorrect. n(ADT) is the total number of training data whose class label predicted by decision tree is correct.

4 Find min(PSVM, PDT), then choose class label from that classifier.

In summary, this method calculates the error rate of each classifier base on the training data and trust the classifier that makes less error will do the classification better in prediction on future data. One exception is about the neutral case since neutral is usually the dominant class in all four domains. Also, it observed that the accuracy of neutral label prediction by either classifier is higher than the non-neutral label prediction.

3.4 Learning Performance Assessment and Prediction

The personalization process in the framework uses a progress monitoring mechanism to validate whether the personalized lab environment is able to deliver effective results. If the personalized results are unfavorable, appropriate revisions must be made to personalization in order to achieve the desired learning performance. Hence, assessment feedback is crucial in this process. In order to achieve a feedback loop, a learning performance assessment and prediction module is developed. This module contains three sub-components and requires a bit of assistance from the instructor. First, a JavaScript program is built to match user input command line with the requirement of a specific lab to monitoring user progress. Second, an online post-lab quiz is constructed. The quiz will ask students 10 questions randomly chosen from a question set developed by the instructor for each lab, so as to obtain some information about students' learning gain. Third, a report submission and grading assistant system are set up to collect students' lab report after each lab session and provide rough assessment and grading advise for instructors and graders. Data collection for learning performance assessment module is much more straightforward. The system

collected command line and syslog frequently to allow the module to estimate student's progress. Then, after each lab session, the system obtained quiz results and report grading estimates to determine the effectiveness of the proposed personalized system. By analyzing output from these modules, the framework constructs a feedback loop to keep revising and improving the performance.

The prediction part of this module takes the output of real-time assessment module and student Behavior Analyzer to estimate students' future learning performance.

The Naive Bayes classification algorithm was used to predict student performance in later semester based on earlier semester result and student's behavior. A Naive Bayes classifier is a simple probabilistic classifier founded on relating Bayes theorem by naive impartiality assumptions. It is easy to build and particularly useful for medium size datasets. Three reasons Naive Bayes model is chosen:

1) High performance when identifying at-risk students

2) Naive Bayes model is quick to build and fast to run, and hence, it makes timely prediction possible in the system.

3) Naive Bayes algorithm is also adaptive to multiclass prediction feature well, which best suits to the students log data sets.

One important part of learning performance prediction is to identify at-risk students early. The prediction model can be used as an early warning system to identify at-risk students in a course and inform the instructor as early as possible. Instructors will then be able to use a variety of strategies to provide at-risk students helps for improving their performance in the course.

3.5 Lab Content Manager

Adaptive lab content manager system contains two-stage generation and utilization in its workflow as shown in Figure 3.2 to generate Knowledge Graphs based on Lab content in the system. It needs to first work out the process to generate the knowledge graph including text data processing, word embedding and the graph structure generation in sections 3.5.1 and 3.5.2. Then three applications closely related to personalized learning are built upon adaptive lab content manager, which includes lab material indexing and searching, knowledge graph visualization, and hands-on lab recommendation, as shown in figure below.

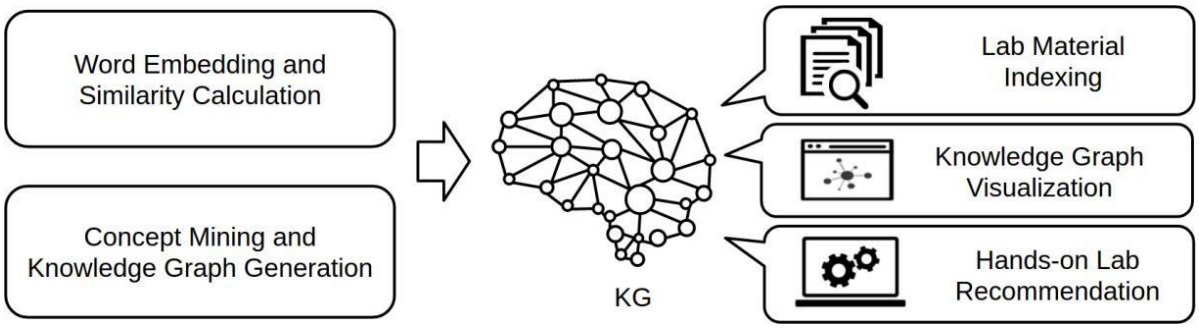


Figure 3.2 Lab Content Manager Architecture

3.6 Lab Content Mining

3.6.1 Word Embedding and Similarity Calculation

For computer to understand natural language and the knowledge and concepts within, words need to be represented in a computer-readable manner. Traditionally, NLP systems treat words as discrete symbols which leads to data sparsity and usually means that one may need more data in order to successfully train statistical models. Word embedding is a set of language modeling techniques to represent word as a vector in a low dimensional space. Using vector representations makes natural language computer-readable, which allows us to perform powerful mathematical operations on words to detect their similarities. word2vec [32] is a two-layer neural network that embeds text. Its input is a text corpus, and its output are a set of vectors, i.e., the feature vectors for words in that corpus. The goal of using word2vec is to group the vectors of similar words together in a single vector space, which help us to connect highly related words (concepts) in the knowledge graph.

The main input of the word embedding module is Wikipedia pages. The English version of Wikipedia database dump on May 1st, 2018, from [33] has been used. A toolkit was developed using Python to scrape Wikipedia pages for the categories in computer security section to acquire more accurate related information. The tool that developed iterates through categories and stores a list of the corresponding information. All main pages in computer security and their related pages in 10 levels of subcategories have been scrapped. There are 7,143 pages obtained under the criteria after removing duplicates. With the processed database dump, several toolkits are designed and developed to train the word embedding model. As a result, there are 4,724,129 unique word embeddings been acquired

which are represented in a computer-readable vector space, of which 1,472,477 are Wikipedia pages titles (concepts). For each keyword, the most similar words can be calculated through the cosine similarity between two vectors. For example, for "DDoS", the top ten similar words generated by the word embedding model are shown in Table 3.1.

Word	Similarity	Word	Similarity
Botnet	0.833809	Honeypot	0.775258
Phishing	0.767333	DoS	0.751166
Denial of Service	0.708786	Spoofing	0.641557
Syn Flood	0.596164	Malware	0.593467
Attacks	0.584982	Crimeware	0.549531

Table 3.1 Top ten similar words of “DDOS”

3.6.2 Latent Semantic Analysis

The NLP tool of latent semantic analysis (LSA) is used to perform the data preprocessing on text data. Latent features are extracted from text data, usually in three steps:

1 The lab descriptions are transformed into a corpus. Data pre-processing is to transfer lab description into a corpus. The techniques for text preprocessing include lower-casing all text data; stemming and lemmatization, which transfers words into their root forms (e.g., using 'connect' to replace the words 'connected', 'connects', using 'good' to replace the words 'better', 'best'); removing stop-words (e.g., is, a, the, etc.); normalization (e.g., using 'iptables' to replace 'ip-table', 'ip-tables', 'ip tables', etc.); removing noise (e.g., digits characters, special symbols, etc.)

2 An NLP's technique of Term Frequency-Inverse Document Frequency (TF-IDF) [81] is used to assign each term a weight from 0 to 1 to indicate the importance of that term to the description as a whole. TF-IDF weights a term by calculating the product of TF and its IDF. The score of TF-IDF shows how relevant a term is throughout all documents in a corpus. For example, terms that frequently show up in most documents are weighted with a low score. In contrast, terms that frequently show up in few documents are weighted a high score since they frequently appear in only a document that carries more relevant information representing this specific document.

3 Latent features are identified using a truncated SVD algorithm [70]. The truncated SVD algorithm finds the most valuable information of the data matrix. It can reduce the TF-IDF matrix dimension by finding similar patterns between terms and documents and combining them into a latent feature vector with a value between -1 and 1.

Each latent feature is a topic represented by specific terms in a document. By following these steps, the system obtains the latent features of lab materials and use such latent features as input to automatically identify which labs are highly correlated than others through similarity clustering.

3.6.3 Topic modeling

The topic model gives an insight into latent semantic topics in a collection of documents and has better predictive accuracy. The inferred topics are more meaningful than using

statistics by providing a hierarchical generative probabilistic model. LSA uses vector representation to represent the text's semantic content. An LSA model replaces raw statistic counts in the document-term matrix with a term TF-IDF score. Then, map these high-dimensional count vectors to a lower-dimensional representation in a latent semantic space. Using LSA, the semantic relations between words and/or documents are represented in the semantic space.

In this study, the system uses the TF-IDF matrix generated by the LSA tool to calculate the value of each lab's input as vector representations. The goal of using topic modeling is to represent features of each lab into a vector space, which help us to connect highly related labs in the knowledge graph. The input of LSA model is lab materials used in our university, most of which are from class lab repository created by instructors in our school, and also labs from SEED lab [82]. The system used these lab materials to build an input dataset for the LSA model, and 130 latent features are extracted. Table 3.2 shows an example of the first 10 topics identified by latent features in this study. The table shows the difference among topics represented by concepts identified. For example, Topic 1 represents labs on attacking through ftp protocol with the concepts of 'ftp', 'file', 'firewall', etc., and Topic 6 represents labs that using Mininet to construct network topology with the concepts of 'mininet', 'switch', 'controller', 'topology', etc. In this way, different topics represent different labs. By computing the 0-1 values that a lab on each specific topic, the system obtains a vectorized representation for a lab to show its value on each topic. Such vector representation captures the latent features of a lab for each topic. K-means clustering algorithm was then applied to group similar labs together. The clustering result is generated based on 130 latent features under comprehensive correlations among these 130 topics for

vector representations. Each vector represents a lab's text data from 180 text inputs for 36 labs, which is identified as a dot in Figure 3.3. It is hard to show the clustering result under all latent features visually; Figure 3.3 shows an example of clusters identified in this study with latent feature value in Topic 1.

Topics	Terms in Topics
Topic 1	ftp, file, linux, directory, packet, attack, firewall
Topic 2	packet, attack, lab, ip, server, dns, report
Topic 3	attack, dataset, python, datum, training, dns
Topic 4	dns, server, attack, attacker, domain, corn, web
Topic 5	vpn, packet, tunnel, interface, datum, program
Topic 6	attack, secret, mininet, switch, controller, cache, topology
Topic 7	web, http, apache, elgg, site, request, ftp
Topic 8	student, lab, vpn, section, firewall, security, vm
Topic 9	vpn, secret, firewall, execution, array, cpu, cache
Topic 10	xterminal, ftp, connection, mininet, attack, tcp, server

Table 3.2: The topics of the first ten latent features

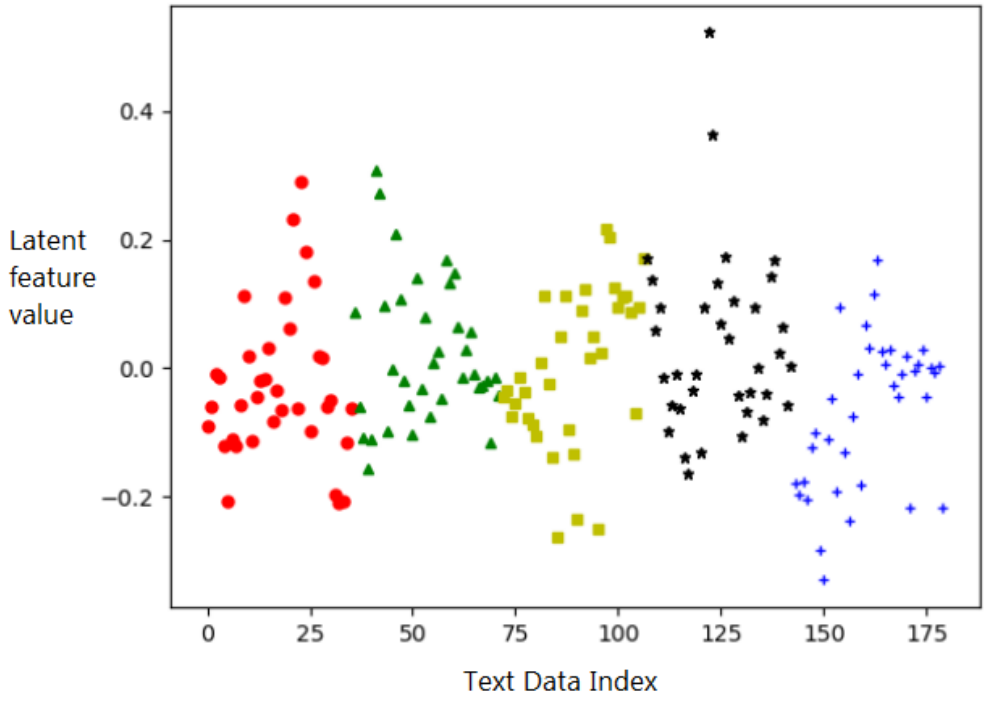


Figure 3.3 The 5 Clusters of Topic 1 Identified by Latent Features

3.6.4 Knowledge Graph Generation

After gathering the word similarities from the previous section, the system is able to generate a knowledge graph in the system. Traditionally, the knowledge graph generation is handled by human experts. The first step is to do manual text analysis and get a list of concepts, represented as labeled points, and a list of links between these nodes. By combining the lists of concepts and links, a small knowledge graph from a single author is then generated, which is called an author graph. The next step is to combine graphs from various authors into one large graph by identifying common points with each other. When

the texts of the nodes deal with the same subject, points with the same label are first identified. Then, human help is needed to identity synonyms for the same concept and connect these synonyms together. One way is to compare the neighborhoods of points. Computing the similarity between two concepts' neighborhood points help us to decide if these two concepts are identical. This method even helps us to detect homonyms, which means the same label but referring to different content. In this case, each Wikipedia page represents a concept and its explanation (which contains knowledge). There are also hyperlinks within each Wikipedia page that links to other concepts. By analyzing the URL links within one Wikipedia page, the system generated a simple author graph. For example, on the DDoS page, there are hyperlinks that linked to Exploit, Trojan Horse, IDS, IPS, Computer Fraud, Botnet, Firewall and computer Virus. With 7,143 pages under computer security category in Wikipedia, there are now 7,143 single author graphs ready to be merged. It utilizes the similarities obtained during the word-embedding process described in section 3.6.1 to further connect these small graphs. Figure 3.3 showcases how to merge graph of 'Firewall' and 'DDoS' graph into one graph. Word pair like {Antivirus, Computer virus},

{Spyware, Trojan Horse} are connected together in Figure 3.4 as their similarity based on word embedding is high. The similarity lower limit is set to 0.8 (while 0 means no relationship and 1 means the two concepts share the same embedding) and connect all node pairs over this similarity threshold. After that, one unified and also highly connected knowledge graph is ready for further utilization.

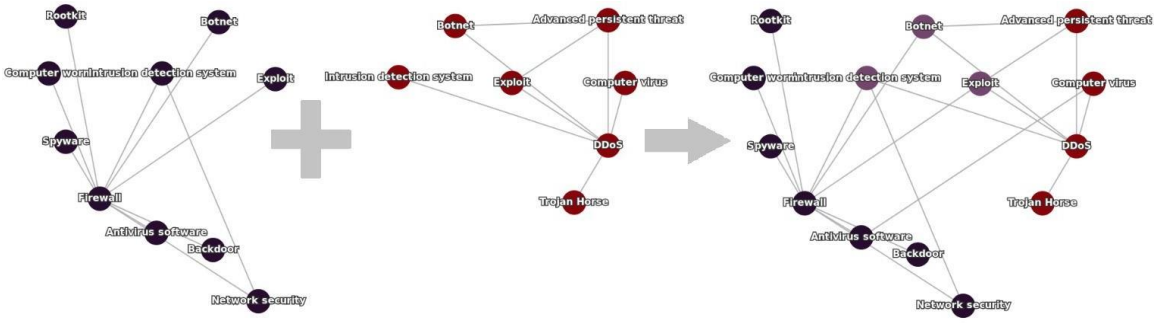


Figure 3.4 Merge two small graph together based on overlap and word embedding similarity.

The threshold 0.8 is used as the lower limit for the following reason according to the experiments: when 0.85 is applied, there are more than 2,000 unconnected nodes, which means these concepts under computer security category are not closely related compared to speaking language words, which is a sign for us to reduce the threshold. There still exist 673 disconnected nodes/small graphs that cannot be included in the main knowledge graph with a threshold as low as 0.7.

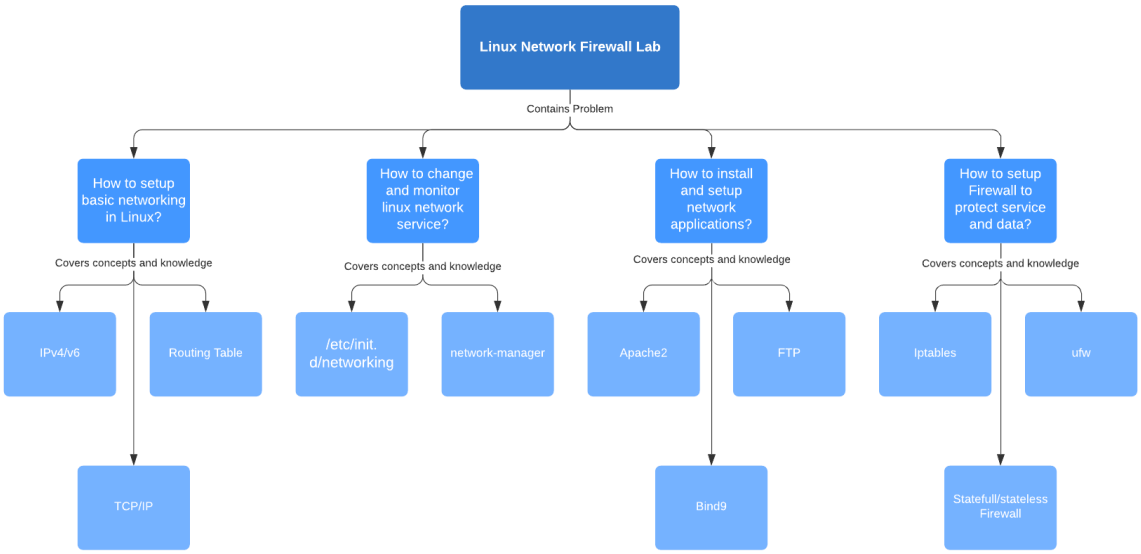


Figure 3.5 Problems and Concepts Mapping for a Single Lab

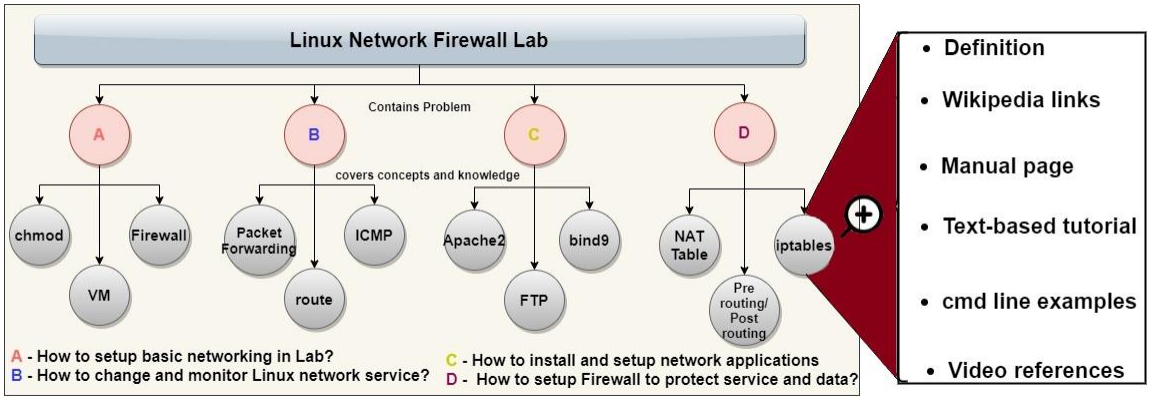


Figure 3.6 Knowledge Graph of Linux Network Firewall Lab

CyberKG is defined as $G = \{V, E\}$, where $V = \{vi\}$, $E = \{eij: (sij, dij)\}$, vi represents a lab, an edge eij includes two measurements: similarity measurement sij and dependency measurement dij. For example, Figure 3.5 shows the graph for a single lab. In the graph, a statement node represents a single lab task, e.g., "setup basic networking in Linux", "setup network application", etc. A statement node can be mapped to one hands-on lab, and each lab is described by a procedure of tasks. And each statement node is connected to a set of concept nodes. Each concept node represents a concept that is required to solving the corresponding task, and its explanation contains knowledge. The document-topic vector representation matrix generated in Section 3.6.3 is used to compute the similarity between the embedded vectors (Ti and Tj) for labs (vi and vj). The semantic similarity ($sij = f\ (vi, vj)$) between labs is computed using the cosine of the angle between two vectors projected in an n-dimensional that corresponds to a topic tk, $k = 1...n$ in the lab vi:

$$s_{ij} = \frac{\vec{T^i} \cdot \vec{T^j}}{\left\|\vec{T^i}\right\| \left\|\vec{T^j}\right\|} = \frac{\sum_{k=1}^{n} t_k^i t_k^j}{\sqrt{\sum_{k=1}^{n} t_k^i} \sqrt{\sum_{k=1}^{n} t_k^j}},$$

where, $Ti \cdot Tj$ is the dot product of the two vectors that represent lab vi and lab vj with n topics, smaller the angle between labs, higher the similarity. The system then constructs CyberKG by measuring this similarity among labs. If the cosine similarity between two laps is over a threshold, their lab nodes are connected in the knowledge graph.

A knowledge graph was ultimately built, which contained 372 concept nodes, 130 statement nodes and 36 lab objects. To compare the quality of the LSA model with the embedding-based approach used in Section 3.6.1, both methods are tested on the same 180 text data from 36 labs used in Section 3.6.3. The system then ran both models for 10 times with the same number of targeted topic number/embedding dimension value of 180 and calculate the Cohen's kappa coefficient [83] (range from 0 to 1, high is better) between the machine learning output and expert knowledge. The embedding-based approach achieves a kappa coefficient of 0.56 while the LSA's result is 0.71. Thus, LSA is able to achieve a more substantial agreement with expert knowledge and provide a solid improvement over the embedding-based approach.

3.6.5 Lab Material Indexing

Within ThoTh Lab, a cybersecurity lab repository is created and made available to instructors and students in our university. Lab design and material from labs of computer

science courses within our school and other high-quality open sources labs like SEED Labs from Syracuse University [82] are implemented. Instructors are able to upload their own new lab materials into the lab repository at any time. All labs in the lab repository are tagged with keywords by matching the lab material with concepts available in the knowledge graph. For example, keywords the system identified in "Local DNS Attack Lab" from SEED lab include: {DNS, bind9, cache, hostname, IP address, LAN, pharming, RFC, rndc, sudo, Ubuntu, Wireshark}. Some of these concepts, like {sudo, Ubuntu} are not directly related to DNS attack, but these are necessary knowledge for each student to finish this lab successfully. Instructors may also edit these concepts before adding them to the lab repository if they think some important concepts were skipped by the system.

The system now gets one lab to N concepts mapping in KG, which allows it to index labs based on nodes in the knowledge graph, and vice versa. As each lab covers at least one node in the knowledge graph, given any two Lab material A and B, the system may obtain their related knowledge graph nodes as the set SA and SB. A similarity of these two articles can be calculated as follows:

$$sim(A, B) = \frac{|S_A \cap S_B|}{|S_A \cup S_B|}.$$

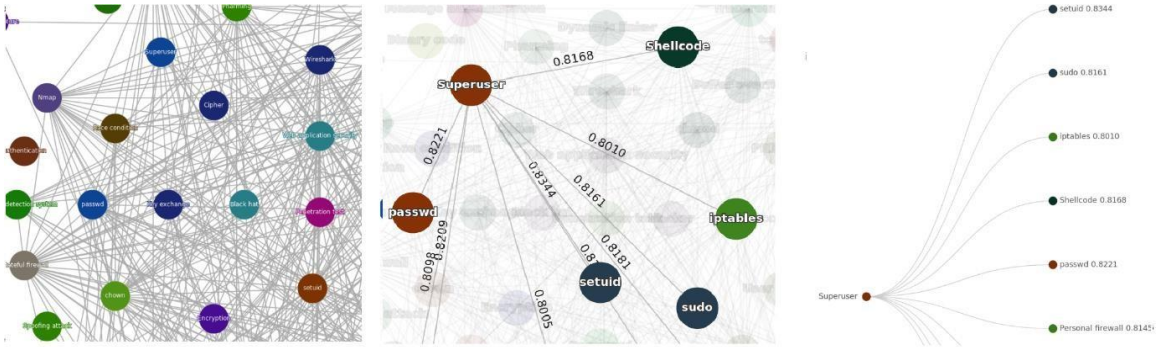
General speaking, the more overlapping between two labs' knowledge graph coverage, the more similar these two labs are. This similarity will then be used as the input of the recommendation module described in Section 3.6.7. Learning material is another component in KG. The system currently linked each node in KG to its Wikipedia page,

which can serve as basic reading material for students. In order to expand the reading material repository, future works including indexing research papers available online.

3.6.6 Knowledge Graph Visualization

With the Knowledge Graph represents in a graph data structure, the next step is to represent the graph in an interactive GUI to empower instructor and students to use it. Since ThoTh Lab itself is a purely web-based lab environment, it is necessary to integrate KG system into the Web UI seamlessly. In this project, E-charts is utilized, which is a web-based visualization library that features a plethora of APIs to creating interactive and dynamic content on the web. The graph was first visualized using three different ways.

First, a full knowledge graph is presented to the user. As shown in Figure 3.7(a). The user may zoom in and hang over nodes in the graph to highlight nodes' neighborhood and gray out unconnected nodes, as shown in Figure 3.7(b). Furthermore, the user may click on one node to generate a tree graph using the selected node as root, as shown in Figure 3.7(c), leaves in this graph can be further expanded. The system also adds the search function to help the user locate concept nodes and index function to show the related labs for each node. The color of the nodes represents the lab it belongs to.



(a) (b) (c)

Figure 3.7 Web-UI for Knowledge Graph: (a)Part of KG (b)Mouse hang-over ’Superuser’ (c)Mouse click on ’Superuser’, which generate a tree using ’Superuser’ root.

3.6.7 Lab Recommendation

Traditional education recommendation systems derive the user preferences from predefined features like user age, sex, educational background, previous grades and/or pre-course survey results etc. The system utilizes the concepts in KG and in the lab materials to recommend labs that suit the needs for instructor or students.

There are two types of students who use the Lab system. The first type is those who are taking a course which uses the lab platform as an instructional tool. Instructors of such courses need to create syllabus and lab planning for the class at the beginning of each semester. The system provides instructors with adequate lab materials within the lab repository. An instructor may provide a list of concepts he/she wants to cover during the course run within KG, and the system will return labs related to these concepts based on the concept-lab indexing generated in Section 3.3. During the course run, the system is also able to identify students at-risk or challenged based on their previous lab grades, quiz

results, and lab activities to make extra lab practice suggestions. Such suggestions turn out to be simple and straightforward, that contains only one lab, which is either the lab with highest similarity (defined as sim (A, B) in Section 3.6.5) to the lab which the student was not able to finish or a lab that covers concepts the student lost most points in their exam or quiz.

The second type of students is graduate students who use the virtual lab platform as a self-tutoring platform for cybersecurity study. They are the target audience of the recommendation module. For these students, an entry-survey was carried-out to check their background in the cybersecurity domain. Then, each student is asked to pick either a set of concepts/knowledge they want to cover or a lab within a lab repository they want to finish independently as their personal learning goal.

The system utilizes the CyberKG and lab materials to recommend labs for instructors and students based on their learning goals and expected learning outcomes.

To achieve that, an entry-survey was first created check students' background in the cybersecurity domain. Then, each student selects either a set of concepts/knowledge they want to cover or a lab that they want to finish independently as their personal learning goal in a lab repository. The KG system first estimates the concept node coverage of a student based on his/her entry-survey results and update these concepts as mastered in his/her personal knowledge graph. The set of mastered concepts CM and the concepts covered by the student's learning goal are defined as CG. After that, KG is able to generate a set of paths PMG between CM and CG using the knowledge graph. Each path P in PMG contains a set of concepts Cp. Combine all Cp together, the system can calculate CP. It is assumed that CP includes all the concepts a student needs to learn and practice in the lab system in

order to achieve his/her learning goal. The last step is to find a set of labs L that covers all concepts in CP. Currently, the system will generate L where each lab in L got high sim (A, B) with another lab in L. This results in a set of labs that shares a lot of concepts between them. When students start working on such labs, they will have the chance to consolidate their current CM while learning the new concepts. L becomes the recommendation to a user. Each time a lab is finished and graded, the system update CM and regenerate L to see if there is an update needed in the suggested recommendation.

Another scenario is when a self-learning student uses Lab Content Manager system without providing any personal data and goal. In such case, Lab Content Manager will not give any recommendation at first. Instead, it will obtain lab activity data when users start doing their first few labs and record their knowledge gained through the lab experience to generate CM for the students. Once enough labs are completed and basic concepts are covered in the user's personalized knowledge graph, Lab Content Manager system will start providing future lab recommendation based on lab similarity ranking calculated and sorted by sim (A, B). By doing these recommended labs, the user will quickly consolidate the knowledge they have acquired and steadily expand their personalized knowledge graph.

3.7 Lab Instruction and Code Storage

To help lab content owners and creators to protect their intellectual property at a very low cost, it is proposed to create a decentralized, transparent blockchain that stores lab instruction and codes while keeps immutable records of copyrights. The system is being designed to keep the following features:

1. The content owner can control easily and always. The concept is totally different from the conventional center operated rights management system. This means owner can control everything. To realize this concept, simple and easy operation would be required.

2. Reasonable security and simplicity can be realized. The conventional Bitcoin system which is the first product to use the blockchain mechanism takes about 10 minutes to mine the Hash value for the calculation. This is because to compete the fastest calculation in order to avoid the pirates. In the case of the digital content distribution, these long mining time will disturb the operation.

Considering a various requirement, the public and private key operation system with blockchain mechanism is being designed carefully. The priority is the usability from the user's point of view. The basic mechanism is very similar with those conventional blockchain application like bitcoin. The most significant difference is that the system will not convey the cryptocurrency as incentive for users. There are two functional stakeholders in terms of the trading. In the case of the digital content distribution model, the content owner is the content rights holder, and the student is a user. The traditional file distribution model requires the authority who handles the right management. The blockchain model has no authority. The user will generate the blocks, all transitions are recorded as a history and all users share these blocks. The structure of a blockchain is that a block that consists of multiple transactions relates to a previous block in chain-like form. To ensure reliability, when a new block is added to the previous block, a little special process of solving a puzzle, called proof-of-work, is needed and this puzzle had to be non-trivial. This process can prevent attackers from forging the blockchain on their own. Another big difference between the existing cryptocurrency and the proposed system is the incentive and the media.

Each miner will consume his own computer resource to get a reward when he finds the conditional hash value faster than any other miners. The proposed system takes the digital content but not the currency. The hypothesis is that the incentive for the miner will be discussed in conjunction with the business model, that is the out of scope of this research.

The biggest merit to adapt the blockchain mechanism to the digital content distribution model is the authentication scheme. The system requires no centralized rights management organization. All participants have all transaction history, in the blockchain. The encryption technology will be adapted to the proposed system also as the same manner with the conventional DRM system. The balance of the decrypt cost and the security level will require the combination of the secret and public key technology. The main target of this system is to show the possibility and potential of the Blockchain-based digital content distribution system. And there are three parties/modules involved in each transaction:

Content owner: The two major function of this module are the permission control for each owned content and upload the content file. The only content owner can control his own content with permission management. The unique characteristic of this system is that the content owner can change the permission anytime; even after the content distributed. From the content owner viewpoint, this anytime-off function is very important, because the contract between owner and users is the limited. For example, the limitation of the content usage is by the expiration, limited number of plays, or some owner's will. When the owner finds any inadequate expression on his works, he must delete it and modify it. The existing content management system is not easy to satisfy this requirement.

Client: Two major applications are running with in user client. One of these is the license control application, which store user identity certificates and compare it with the rights

information from the Blockchain. The other part is a controller module that control the client based on the result. The client could access data only if the license certificated.

Sorting server: This is the main module of this system. The mining function are, to generate the new block which include the rights information, to add the nonce with some calculation, and to broadcast the new generated block on the network.

3.7.1 Storage of Metadata and Digital Content

There needs a storage model that is not based on trust between client and host. All client private data, including filename, date, and other metadata, must be encrypted before any transfer takes place from a client's computer to the cloud. There can be no centralized point of attack using political or legal attack vectors. All incentive payments for both resource providers and consumers will be automated and made in a pseudonymous cryptocurrency. It must design the nodes and network in an extremely secure manner as neither the lines of communication nor the nodes themselves can be trusted. Nodes on the network must collaborate to achieve the level of redundancy and performance of current centralized networks. Furthermore, the software must run by itself and without manual intervention in both practical and economical aspects.

Thus, a distributed file storage system (IPFS) [40] is used to account for two key regulatory considerations: (1) uncertainty with regards to what anonymization techniques are legally sufficient to transform the copyright data into anonymized data and (2) to minimize both the transmission and storage cost.

The IPFS is a peer-to-peer distributed file system based on content-addressed hyperlinks. As such, it takes files and manages them based on their content, storing them and tracking their version using a generalized Merkle directed acyclic graph. These Merkle trees, or hash trees, allow secure verification of the contents of large data structures, using cryptographic hash functions that map data of arbitrary sizes to data of a fixed size.

Advantages of this technology include: (1) data stored on IPFS are not automatically distributed between all participants and only shared in the case of a request, (2) IPFS nodes are able to delete specific data at any given point in case of a request, and (3) it is easy to prove whether an input will result in a given hash, but incredibly difficult to derive the input from a hash.

In this design, the aggregated files were uploaded into a private IPFS cluster via a writable IPFS gateway. Every participant of this network was publicly known and can be held accountable in the case of noncompliance with a data deletion request from the data subject. Therefore, this private setup allowed for the specification of the number of backup copies in the network, and for the definition of automatic rules, such as when to delete data in the case of a content owners request. To link IPFS transactions to the authenticated and undeletable transactions, hashes of the IPFS content were then uploaded to the blockchain. For simplicity, SHA256-256 hash function is used with Base58 encoding, which is the default hash function of IPFS.

After the file is encrypted and uploaded to IPFS, the SHA-256 hash is identified, which serves as both a unique identifier and a way to detect file tampering. If any alteration of the file occurs after it is uploaded, the hash will be different. This fact is used in the underlying platform IPFS so the network can spot check the files without having to access them

directly. A client can also use hashing to ensure that received files are authentic. The hash will be stored in a blockchain entry along with the storage locations of the file used to generate the hash. All metadata inserted into the blockchain can be protected from unauthorized reading and copying using public key encryption. Because all data and metadata entering the network is encrypted, and the system can verify data through hashing, malicious entities cannot spy on, fake, or modify the data.

CHAPTER 4

EXPERMENTS AND RESULTs

This section divided the evaluation of this work into several major parts:
evaluation of learning style identification results, evaluation of learning performance prediction results, evaluation of lab recommendation results, evaluation of instructor and student feedback.

4.1 Experimental Setup

Four phases of experiment have been conducted. First, a field experiment of the ThoTh Lab was conducted on an upper-undergraduate-level class during the 2019 Spring semester in Arizona State University. This particular course is on network security and involves 5 hands-on labs about practical network configuration with the usage of basic network security concepts, case studies on attack and defense, and useful tools for reconnaissance and penetration. 103 senior undergraduate students registered the course, and all of them finished the ILS Questionnaire before the first lab to provide an estimation of the ground truth of their learning style preference. During the semester, each student was asked to finish first three labs in an environment on their own personal computer, then two more labs in the proposed personalized virtual hands-on lab environment. All the five labs were based on the same topic and the same contents were used in the previous semester, with only minor modification to prevent cross semester plagiarism. Additional lab materials in video and picture and different formats are made to enable adaptive learning content for

different learning style. For the first three labs, all students were asked to record how much time they spent on each lab. For the other two labs, students' activities were recorded online and inside the virtual machine. At the end of the semester, all 103 students were asked to finish an exit survey.

Second, an experiment using the upgraded Lab Content Manager module was conducted in a graduate-level network security class during spring 2020 semester at Arizona State University. This class involves three hands-on labs for computer networks security. 23 graduate students took the course, and all of them finished the pre-survey before the first lab to provide an estimation of their network knowledge backgrounds. During the semester, each student was asked to finish three labs in the virtual lab platform. They were also asked whether they wanted to volunteer in the research practice, and nine students participated. These nine students set their own learning goals on the knowledge graph and then got the recommendation of labs as an outcome of the Lab Content Manager system. They continued to work on these labs, and 8 of them finished all recommended labs. All students' activities during the labs were recorded in the browser end and inside the virtual machine they used. At the end of the semester, all 23 students were asked to finish an exit survey, where those nine volunteers got extra questions to answer. In the exit survey, the student satisfaction on the hands-on virtual lab platform has been analyzed and they were also asked about their opinion on Lab Content Manager system.

Third phase of experiment using CyberKG was conducted in a graduate-level network security class during Summer 2020 at Arizona State University. This class involves five hands-on labs for computer network security. Forty-three graduate students took the course, and thirty-four of them finished the post-course survey at the end of the semester.

During the semester, all forty-three students were required to first finish three labs in the virtual lab platform as part of their course evaluation. They were also asked whether they wanted to volunteer in this research practice, and thirty-eight students from the class participated. These thirty-eight students set their own learning goals on the knowledge graph and then got the labs' recommendation as an outcome of the CyberKG system. They continued to work on these labs, and thirty-four of them finished all recommended labs. At the end of the semester, all these thirty-four students finished this post-course survey. Twenty-three of the students strongly agreed that this lab-based learning approach motivates them to learn computer science security. Further, thirty-one students enjoyed this lab-based learning experience.

To construct the exit survey (APPENDIX D), the system follows the Instructional Materials Motivation Survey (IMMS) [84] to identify student motivation when doing this problem-based learning lab. IMMS is widely used in previous studies on education to evaluate students' motivation to work with technology [85] or a web-based course [86]. These survey questionnaires evaluate students' motivation from eight areas, including course overview, student's attention, the relevance of learning materials, the relevance of projects, student's confidence, student's satisfaction, and lab-based learning through role-playing and lab-based learning in general.

Last phase of experiment is situated in a week-long professional development event aiming to introduce researchers, educators, and other working professionals with content and domain knowledge of cybersecurity and help them develop problem-solving skills for cybersecurity. During the event, participants were asked to finish 2 to 4 hands-on labs

recommended by the system. 9 of the participants were recruited from those who had completed the professional development event as interviewees. The interviewees were a random sample, and every trainee who expressed their interest in this research were all interviewed and included into the analysis. Among the trainees, there are 3 female participants and 6 male participants, where 2 of them is a master student pursuing a master's degree in computer science while the rest 7 of them are with a graduate degree in CS. There are 3 of the trainees working as professionals in the IT industry, while the rest 6 trainees are still studying and doing research at their universities.

4.2 Data Collection & Results

4.2.1 Phase One and Phase Two Experiments Results

During phase one and phase two of the study, various types of data were extracted from the interactions between the student and the Web-based education system. The data the system was able to record, and measure generally depends on the capability of the personalized virtual hands-on lab system. Thanks to the system's web and cloud nature, it is not difficult for us to capture all the web page activity of each student and Linux system log from each virtual machine they have used. It listed two uncommon features collected during students' lab period and the motivation of choosing them. The first feature is hint link access counts. There was a hint bottom next to each section of Lab 4 content and students are allowed to click them to get help on their next lab task. The more times they click on the hint bottom, the more detail the hint will be. There are 3 levels of hints for each hint bottom. This feature is designed on purposely to identify students who have hard

a time understand each lab task and finish tasks by themselves. The second uncommon feature the system collected is video viewing timestamps when students start and stop view guidance videos in lab content. It was collected in order to find those students who prefer visual learning material over traditional text-based guidance. Experiment Result

Learning style identification: The system used the distribution of student learning style identified by ILS questionnaire as the verification data for the learning style classifier (shown in Table 4.1). System then used students' learning behavior log from the 4th lab of the semester to train and test the three classifiers in learning style classification module, as shown in Table 4.2. The test used 10-fold cross validation method to calculate the accuracy rate of the classifier output in each learning style category.

Learning Style	# Of Students	percentage	Learning Style	# Of Students	percentage
Visual	30	29%	Active	20	19%
Neutral	59	57%	Neutral	61	59%
Verbal	14	14%	Reflective	22	21%
Total	103	100%	Total	103	100%
Sensory	31	30%	Global	24	23%
Neutral	40	39%	Neutral	51	50%
Intuitive	42	41%	Sequential	28	27%
Total	103	100%	Total	103	100%

Table 4.1 ILS Questionnaire Results

As expected, using the ensemble of classifiers results in an acceptable accuracy in all four dimensions, as it always selects the classifier with lowest misclassification rate in a

particular category. The classifier's major performance gain is on the neutral label prediction, compared with either SVM or Decision Tree method. This is directly caused by the special design of the label prediction algorithm that gives neutral label more weights. However, the classifier demonstrates limited performance in the Active/Reflective category, possibly due to the lack of high-quality features for that dimension, i.e., the selected features are not well correlated to active and reflective learning style. On the other hands, thanks to high-quality features like video viewing timestamps, the classifier works well in the Visual/Verbal category. The ensemble of classifiers performs worse than individual constituent classifiers in a few special cases. For example, the ensemble method returned an accuracy rate of 0.687 when identifying sensory learner, while DT returned 0.75 and SVM returned 0.792. Still, the performance improvement of the ensemble of classifiers in identifying neutral learner in all dimensions is more than enough to cover the loss. Table 4.3 shows the comparation of accuracy between our approach with current state-of-art data-driven Learning Style identification methods.

Learning Style	Classification Accuracy	Learning Style	Classification Accuracy
Visual	75.0%	Active	64.5%
Neutral	83.6%	Neutral	69.7%
Verbal	77.2%	Reflective	52.4%
Total	80.6%	Total	68.0%
Sensory	68.7%	Global	66.7%
Neutral	80.4%	Neutral	91.5%
Intuitive	69.6%	Sequential	71.4%
Total	74.8%	Total	81.6%

Table 4.2 Learning Style Classification Accuracy

Algorithm	V/V	A/R	S/I	S/G	Average
DT/SVM	80.6%	68.0%	74.8%	81.6%	76.3% with S/I 79.0% without S/I
NN[94]	72.7%	80.2%	74.1%	80.5%	77.4%
Deles[95]	76.7%	79.3%	77.3%	73.3%	76.7%
Bayesian[96]	NA	58.0%	74.0%	63.0%	66.0%
NB tree[97]	53.3%	70.0%	73.3%	73.3%	67.5%

Table 4.3 Learning Style Classification Accuracy Comparation

Learning performance prediction: the distribution of students' final lab grades and final course grades of the semester were used as the verification for the system. Based on the grades, performance distribution of 103 students is presented as Good (23), Average (30), Below-Average (17) and At-Risk students (10). Students' grades from lab 1-3 and learning behavior log from the 4th lab are both used to train and test the Naive Bayes classifier in learning performance prediction module. The system used 10-fold cross validation method to calculate the accuracy rate of the prediction output for each category of students. Table 4.4 shows the results.

Student Category	Prediction Accuracy
Good (Grade A or Above)	82.1%
Average (Grade B and B+)	69.8%
Below Average (Grade C, C+ and B-)	81.0%
At Rick (Grade D and below, no Grade C-)	90.9%
Overall	77.7%

Table 4.4 Learning Performance Prediction Results

With the benefits of fast training on small data set, the performance prediction model still yields acceptable overall accuracy rate, while providing over 90% accuracy on at-risk student detection. As discussed earlier, identifying at-risk students is the major goal for

most learning performance prediction models. It is important for instructors to identify at-risk students in order to provide timely interventions. Thus, Naive Bayes model fits the predication goal well.

lab recommendation: An example of the recommendation process for one student is shown below. At the beginning, the system estimates that his knowledge coverage CM contains Linux command line, Linux Network and Firewall, and he picks the learning goal CG containing only SSL Session Hijack. Then KG generated CP for him as shown in Figure 4.1. Based on CP, a L of five labs were recommended to him:

(1) Linux web service lab, which covers two concepts in CP (blue squares),

(2) Linux firewall lab, which covers two concepts in CP (green squares),

(3) Packet Sniffing lab, which covers three concepts in CP (red squares),

(4) IP and port scanning lab, which covers three concepts in CP (purple squares), and

(5) SSL Session Hijacking Lab, which covers four concepts in CP (yellow squares).

It is noticed that the process and result reasonable enough but were not able to do quantitative analysis on the recommendation result. Student feedback on the result can be found in later section.

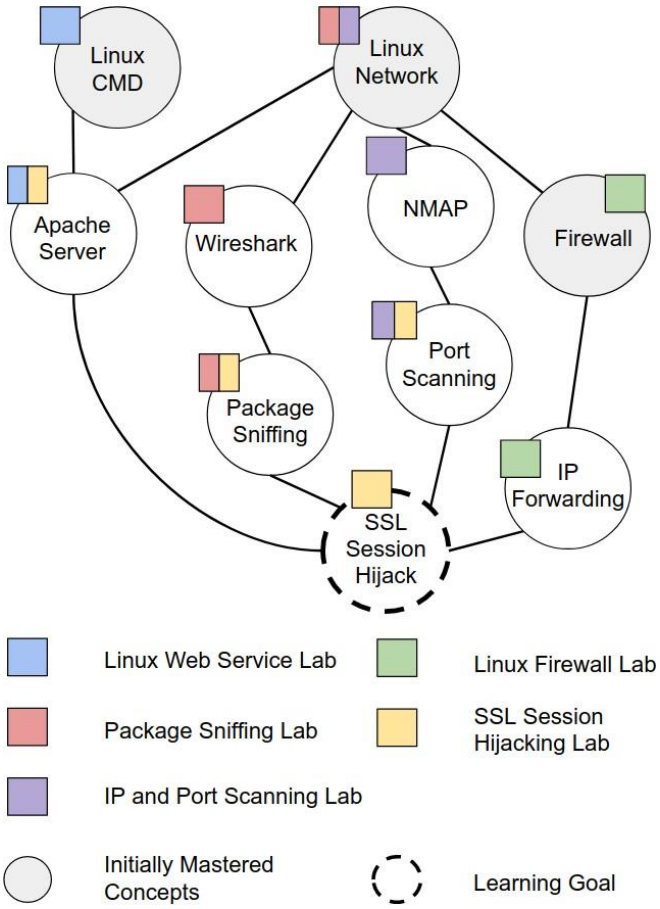


Figure 4.1 Sample Lab Recommendation Process

Student Learning result and feedback: A case study was then conducted using predicted learning style labels shown in above section as initial input for adaptive learning content management module for the 5th labs, the feedback from learning performance assessment module and final lab grades are used to assess the proposed system's effectiveness on students' learning performance. The case study result shows that majority of students achieved better grades after the utilization of personalized lab materials for their individual learning style for Lab 4 and Lab 5 (shown in Figure 4.2). Among the 18 students whose performance was negatively impacted by personalized lab materials by more than 3%, all

of them are from At-Risk category, and 17 of them withdrawn from the class during the study, resulted in 0 grades for lab 5. Compare with the original student performance distribution, it shows that the personalized lab materials provide more positive impact on students with better performance. The average grades of Lab 5 also show improvements when be compared with the same lab from Spring 2018, which also uses the same virtual lab system, but without the personalized framework (shown in Figure 4.3). Interestingly, students were inclined to spend more time on virtual labs compared to labs running on their own computers, which can be interpreted as a sign of improved engagement in the lab.

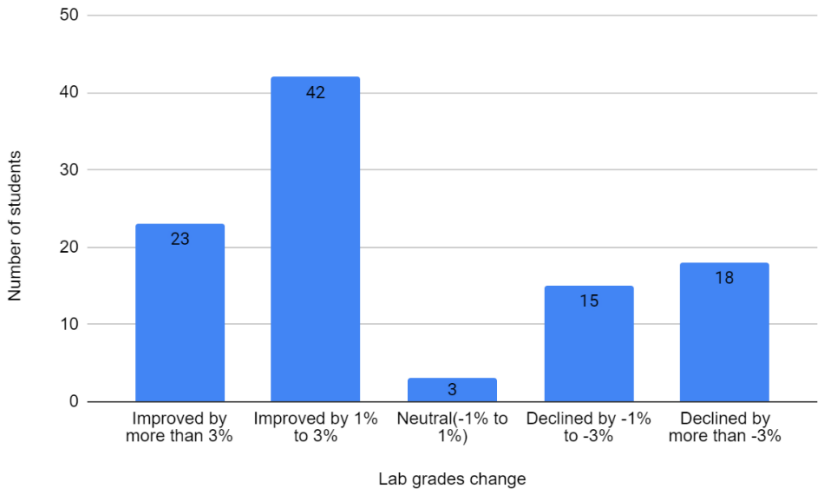


Figure 4.2 Effect of personalization on student learning outcome.

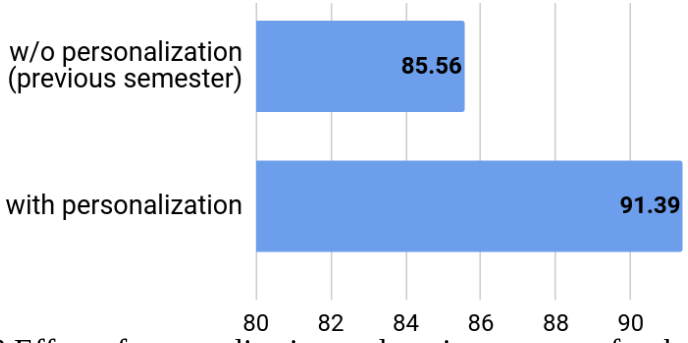


Figure 4.3 Effect of personalization on learning outcome for the same lab

(From different semesters)

In the exit survey （APPENDIX C）, the student satisfaction on the hands-on virtual lab platform has been analyzed and they were also asked about their opinion on Thoth lab system. Answers in exit survey on a scale of 1 to 5, 1 being totally disagree, 5 being fully agree.

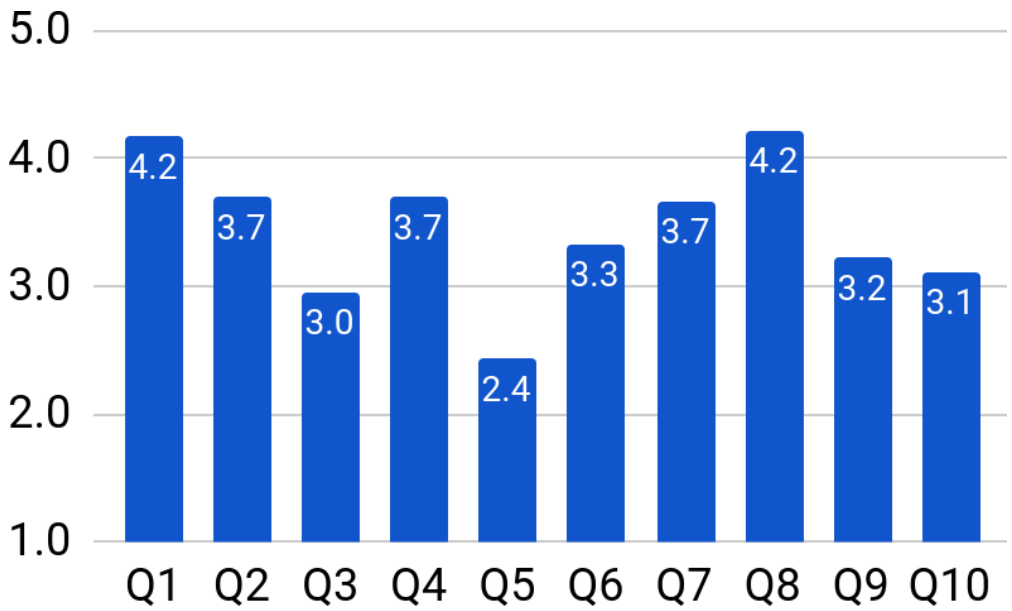


Figure 4.4 Average score of questions in the exit survey.

Figure 4.4 shows the average score for each question in exit survey. Majority of the students left a positive comment after using the virtual lab platform. While the estimation of student knowledge coverage on quiz is not accurate (Q3), it improved significantly at the end of class based on user activity log (Q4). Among the 9 volunteers that utilized the lab content system for learning recommendation, 6 of them agreed that the recommendation is highly related to the topic they pick (in Q7). The survey results also shows that majority of students confirmed the usefulness of the recommendation for hands-on labs (Q6), and the system had a positive influence on their learning attitude during the semester (Q9).

4.2.2 Phase Three Experiment Results

An example of the recommendation process for one user is shown in Figure 4.5. Based on the entry survey result, the user's initial knowledge coverage contains {Linux command line, set up Linux network}. It picks the learning goal of {setup SDN Firewall} only. Then CyberKG generated five recommended labs for him in sequence, as shown in the Figure 4.5. The five labs, in sequence, are: (1) Lab 1, Linux network Lab, which covers three statements (green boxes in figure) and demand basic computer network knowledge. (2) Lab 2, MiniNet SDN sandbox lab, which covers two problem statements (blue boxes in figure), this lab requires the user to set up a MiniNet SDN environment, in which the user will set up firewall later. (3) Lab 3, POX Controller Lab, covers three problem statements (red boxes in figure) and covers how to set up POX as an SDN controller to forward traffic. (4) Lab 4, Linux firewall lab, which covers problem statements (yellow boxes in figure), this lab tests user's knowledge about network firewall and its usage. (5) Lab 5, OpenFlow Based Stateless Firewall Lab, which covers three problem statements (yellow squares in the figure), including the user's learning goal of setting up an SDN firewall. Notice that, only Lab 2, the Mininet lab, is optional, as other labs do not directly require it. But, since the Mininet lab gives users a better understanding of the SDN environment, both are still recommended.

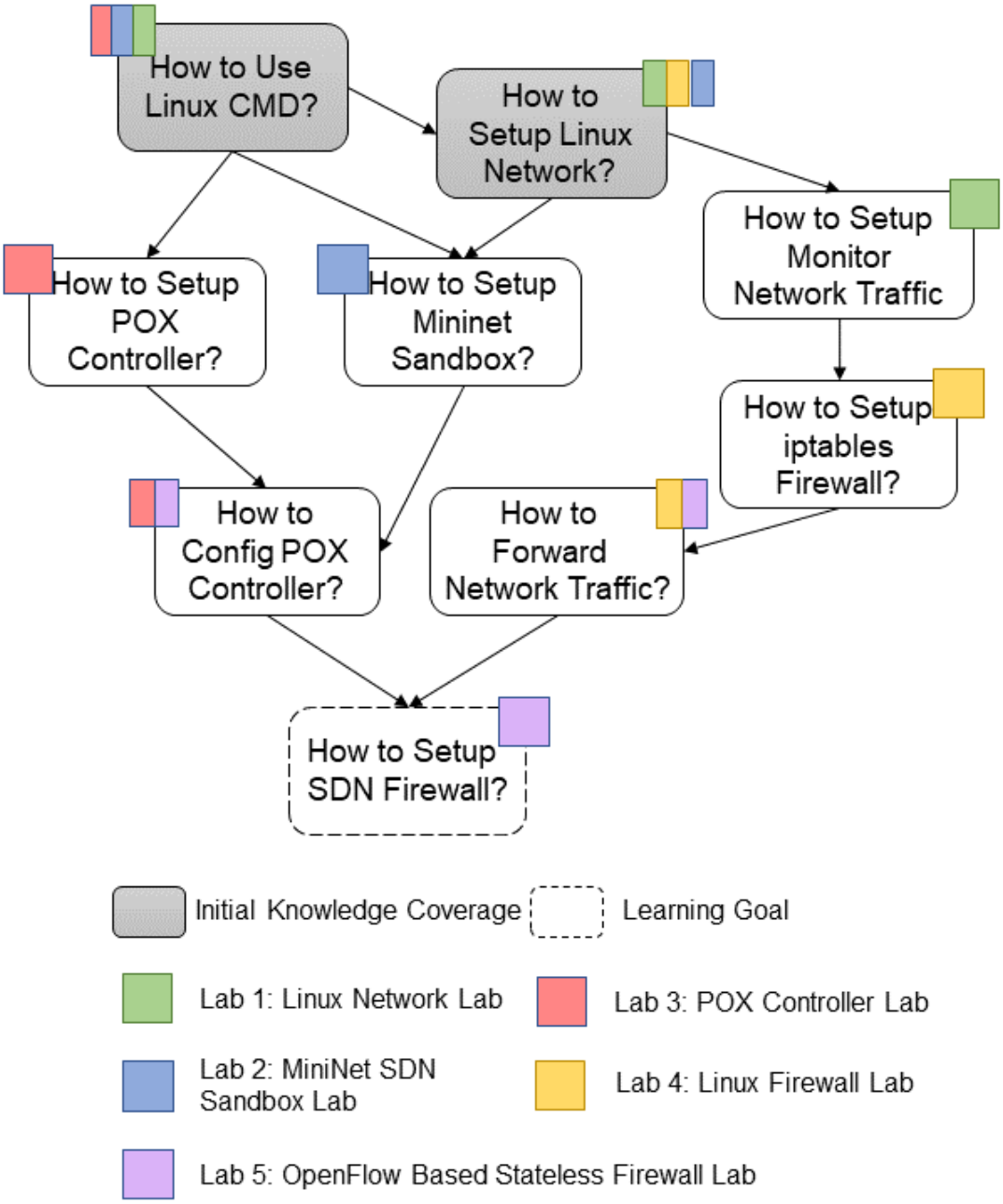


Figure 4.5 Lab Recommendation Process by CyberKG

Figure 4.6 shows each question's average score in Phase 4 post-course survey on lab-based learning. Two questions (Q17 and Q26) are asked as negative questions, so the system transfer the score into a positive score when counting the statistical results. This score shows that most students confirm that this lab-based learning positively impacts their learning attentions (average score = 4.0) and confidences (average score = 3.7). They are satisfied with this lab-based learning approach (average score = 4.2).

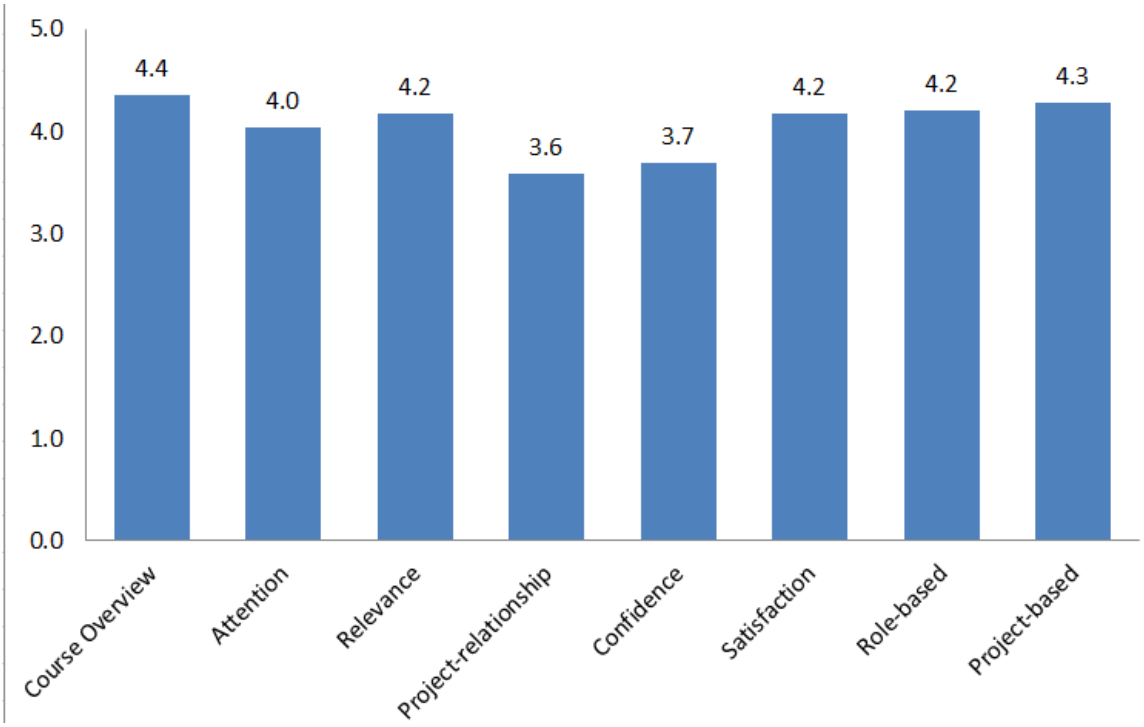


Figure 4.6 Average Score of Questions in Each Area in the Exit Survey.

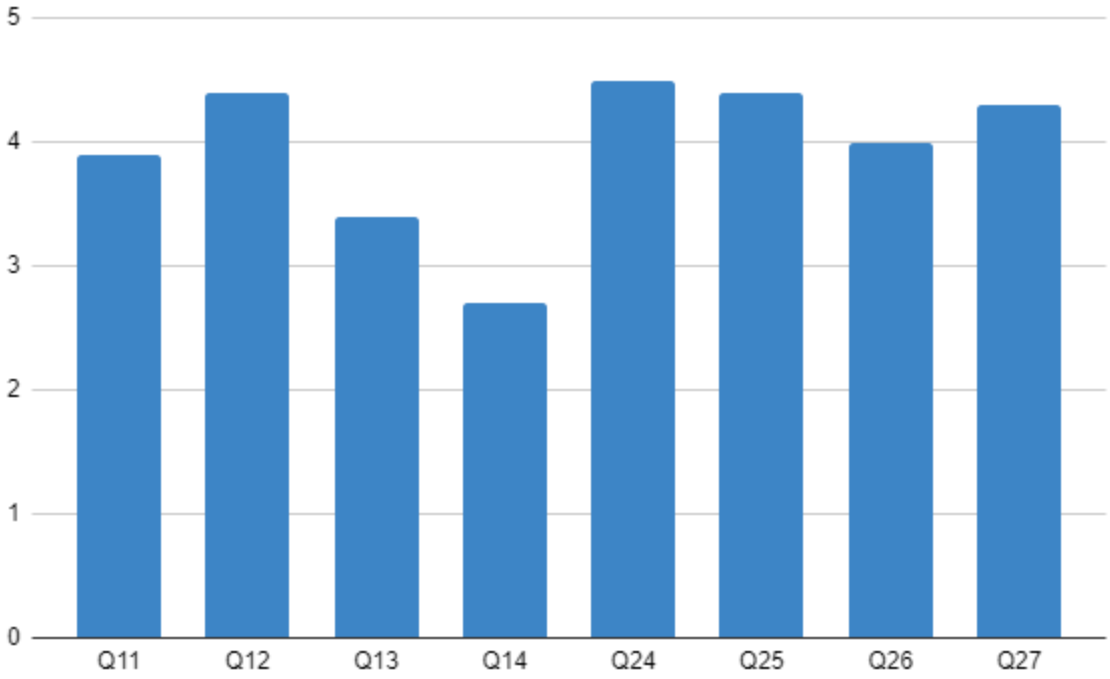


Figure 4.7 Average Score of Questions about Lab Relationships and Lab-based Learning

Specifically, feedbacks were collected from students to evaluate their perceived lab relationships in this case study. Figure 4.7 shows the average score of questions Q11 to Q14 in this area. Lab 1 is a background lab about Linux networking and firewall setup, Lab 3 is SDN security labs. Lab 2 is about SDN network, it is a recommendation generated by CyberKG based on topics and concepts of Lab 1 and 3. Q12 result shows that students strongly agrees that CyberKG recommendation is highly related to Lab 3. Lab 4 is also picked by CyberKG, not only base on topic from Lab 1 to 3, but also based on each student's personal learning preference this round. Q14 result shows students agree that Lab 4 topic is clearly distinguishable from other labs.

For lab-based learning, Figure 4.7 shows the average score of Q24-Q27. It shows that students strongly believe that lab-based cybersecurity instruction enhances their learning skills and leads them to spend more time studying. They think this lab-based learning is better than traditional learning and have a good learning experience under this learning environment.

4.2.3 Phase Four Experiment Results

Every trainee who expressed their interest in this research were all interviewed, and all interviews were included in the analysis. The interview was semi-structured with some pre-selected questions. During the interview, all interviewers are flexible to dive deeper into any questions. They are encouraged to provide further context or relevant information when answering the listed questions. All interviews were conducted virtually over Zoom. The interview started with a demo. Each trainee shared their computer screen and showed a list of lab output/results for a specific lab they finished during the professional development event.

During the demo, the trainees were also asked to explain the results in their own words so that interviewers could better understand their thinking process. The demo was included to observe student behaviors when using the PBL lab system live, which would offer another layer of analysis and rely on their answers to the follow-up questions, especially to analyze problem-solving behaviors during the lab process. After the demo, the interviewers first asked all of the trainees if they have any questions or recommendations about the PBL lab system with KG guidance during their training. After that, the interviewer went over a list of pre-selected questions about the student's experience with the PBL Lab system with KG

guidance. These questions were related to the research question the research aimed to investigate. Each interview takes approximately 15 mins, and all interview sessions are recorded for analysis later.

To analyze the collected data, the transcripts of recorded interview audio files are coded. It follows the open coding approach that used in the previous study [87] to analyze the collected data. The open coding method is the analytic process that attaching the concepts (codes) to the observed data and phenomenon in qualitative data analysis [88].

In the study, two authors coded an interview transcript independently and then discussed them together under several rounds to ensure the standardized coding framework. Then, the coders coded the collected data independently and interpreted the participant's responses by considering the semantic information of the entire interview. After coding all transcripts, the participants' comments are extracted to address the research questions. The analysis results are based on 10 codes representing participants' feedback on using the PBL lab system with KG guidance. Table 4.4 shows the list of codes for each research question category. The participants' learning behavior using the PBL lab system with KG guidance was explored by analyzing the relevant codes from the participants' interview transcripts.

For the motivation part, participants provide the purpose of using this PBL lab system with KG guidance that mainly focuses on creating a cybersecurity simulation environment, completing the group project practice, and supporting the learning experience. For instance, I1 said that “it's convenient for me to deploy any virtual infrastructure remotely”, and the senior professional in industry I2 stated that “(It is) a platform or tool which can quickly deploy, simulate, and verify my network and security architecture design easily.” A6 also highlights the purpose of using this system “to practice real-world problems that

are related to cybersecurity and network security." Differently, the A2 and A3 are motivated by the functions that support group projects. A2 said that "Is it a passion that I also use those machines and in some of my research for work, where we need to work together as a group." Additionally, A5 and A1 emphasized the usefulness of the KG function as "the Knowledge graph is a neat add-on to the learning experience. The mapping on related concepts and useful information at each node really helps to frame an overall concept map better", and "the knowledge graph is also helpful... and provides a lot of resources for me to master this knowledge."

Categories	Codes
Motivation	Perception-Purpose
Problem based learning experience	Behavior-Explore Knowledge and Skill Set under KG Guidance Behavior-Solve Problem under KG Guidance Perception-Confidence Level of Solving Problem Perception-Awareness of Cybersecurity
User satisfaction	Perception-Supports on Real Life Problem-solving Perception-Supports on Learning Cybersecurity Knowledge Perception-Resource

Table 4.5 Coding for Interview Data

Take-away for motivation part: The interview results suggest the similarity and difference between the professional participants from industry and academia on the motivation of using the PBL lab system with KG guidance. All participants expressed their perception of this system that is easy to use and useful. According to the theory of IT acceptance [89], the perceived usefulness and ease of use are the key factors that are positively associated

with the usage of anew IT system. Thus, this PBL lab system with KG guidance might be accepted well by a broader range of users based on the findings in this qualitative study. Additionally, it is noticed that the different needs of using this system based on the role of participants, where the participants from academia emphasized the need of supporting collaborated work that is usually for a course study purpose, and the participants from industry focused on the functions that can support deploy a project. There is not a significant difference among gender in the motivation of using this system. These findings will guide future development on this PBL lab system with KG guidance to better address the needs of professional trainees.

For the PBL experience part, the participants in academia and industry all showed their behavior on exploring the new knowledge and skillset in the system, which are guided by the KG function. For example, I1 said that "knowledge graph provides all this information in a consolidated fashion in one single place." I1 also explained that "the knowledge graph basically helps students understand that correlation between different skills ... I can basically trace those dependencies and learn these in an organized manner."

Regarding how to solve a problem under the KG guidance, it is identified that the KG function gradually guides the trainee exploring from the basic level to the advanced level of knowledge to solve a problem. The representative quotes from A3 are: "I hardly understood the Linux environment and how to do the setup ... it (the KG) shows that what commands you should use for setting up a firewall, what does a firewall means, and it also gives a description. And then, it gives a link where you can go and watch, ... and then all the related... when the tasks were given...it was super easy because you know that

knowledge graph actually have been traveled from the basic level to the further advanced level ... and gradually improve my knowledge in that area."

Regarding their perceptions on the confidence level of solving a problem, most participants felt the lab practice in this study was at a moderate level and felt confident on solving a problem in this case study, where the KG guidance efficiently supports participants in learning cybersecurity knowledge and skill. The key factors that affected the choice of confidence levels were, as stated byA5, "the visualization of the related terms across concepts helps the user in identifying new concepts to learn... KG brings all the relevant information in one place for students to enable effective learning." Only one participant (A2) reported that he might not be so confident at the beginning, and A2 said that "I ended up retaining and learning more than I thought I was, so it was very beneficial." A3 explained how to build up the confidence gradually under the KG guidance as: "before I started it, I wasn't that much sure, like say 20% or so, but after reading the material once ... it was boosted my confidence to around 50-60%, but when I actually did it... go back to the material again ... followed each step carefully, I think I could do most of the part." In addition, all participants agreed that their awareness of cyber-security was increased. Specifically, the industry participants suggested that an advanced level of KG function that including at-tack scenarios could better support the needs of industrial practice.

Take-away for PBL experience: The interview results reveal that the KG function facilitates exploring and obtaining knowledge, organizes the correlated skills for tasks, and effectively supports learning and problem-solving processes. Based on the results, the design of KG function well organized the knowledge/skills covered in a task into a big picture, for example, as shown in Figure 2, which prevented the trainee from being trapped

in microscopic perspectives with an isolated problem/knowledge units and guided the trainee to deliver a comprehensive big picture for the target task. The existing study on cybersecurity education [90] also proves that the KG's multi-layer, multi-dependencies design can help to build a knowledge network instead of isolated knowledge units. The participants suggest an advanced knowledge graph, which covers more cybersecurity concepts, dependencies, and cybersecurity attack scenarios. In addition, it is noticed that the behavior (as stated by A3) that mapping the cybersecurity knowledge learning with the hands-on problem learning contributes to increasing the trainee's confidence level. According to the existing study, the decrease rates of newly acquired knowledge are lowered down by consolidating hands-on learning with cognitive learning [91]. Thus, this PBL lab system with KG guidance might have an advantage in supporting trainees' earning success in the long term, which can be possible future work. All the findings from the interview bring up future work of a more accurate way of gauging the trainee's PBL in this system.

Regarding user satisfaction, all participants preferred using the PBL lab system with KG guidance for other projects in the future. The participants from the industry emphasized the supports for solving real-life problems. For instance, I2 said that "the labs allow me to easily design and deploy a small to mid-scale network security architecture. I can use the deployment to verify my design for feasibility and give it a small-scale performance test." Additionally, I1 said that this PBL lab system with KG guidance "definitely helps understand the different ways an attacker can enter your network, and what are the actions you can take to defend against those kinds of attacks, ... so it definitely helps in improving the security of the data as well as infrastructure in my day-to-day job setting." The

participants from academia under-lined the need to learn cybersecurity knowledge and resource since such supports could help them better prepare cybersecurity skills for the job market as A4 said that “as the Cybersecurity concepts are understood and visualized better with solving practical problems around the concepts. This also prepares students for better practical engineering jobs associated with graduation.

Take-away for user satisfaction part: This part of the interview focuses on investing in the desire for system usage. The participants in this study showed differences in the interested cybersecurity tasks. The findings reveal that the trainees perceived a good performance of tasks by using this PBL lab system with KG guidance. According to the existing study, a fit between technologies and users’ tasks can enhance the task performance [92], and then motivates to use the system for learning [93]. These findings well guide future development of this system on generating a personalized knowledge graph that better supports the individual trainee’s task.

CHAPTER 5

CONCLUSION AND FUTURE WORK

This chapter concludes the dissertation by summarizing the contributions of the work and highlighting the future directions.

5.1 Conclusion

In this dissertation, I created a personalized learning framework in a cloud-based virtual hands-on lab platform for computer science education. The framework was able to automatically extract learning style from an online hands-on virtual laboratory platform for computer science education using data mining algorithm. The framework also enables the personalized learning feature of the platform, which will adopt to each different learning and greatly enhance students' learning efficiency. To improve the accuracy of learning style detection, I adopted a data mining classification method that combines two disparate data mining algorithms, SVM and decision tree. This dissertation revealed that the proposed combination method demonstrates better performance than both single classifiers. This dissertation also proves that it is possible to identity learning style of students purely based on their learning behavior not only in LMSs, but also in online laboratory environment given the right data mining tools and data collection method.

This dissertation also describes my efforts towards creating a knowledge graph to represent concepts and their relationships in the cybersecurity domain. The knowledge graph is intended to provide an organized knowledge base that incorporates information from a large variety of data sources including Wikipedia pages and instruction materials, which includes all relevant concepts within the domain for educational usage. Such knowledge

graph can be then utilized e-learning platform like the proposed personalized learning framework to test it. When using the knowledge graph as a recommendation/guidance tool for students, the case studies proves that the prototype system was able to meet students' expectation when making the recommendation.

The framework also applied Problem-based learning on the proposed personalized cybersecurity lab environment and created a knowledge graph as PBL guidance for learners. Each trainee's problem-solving process was observed in the proposed framework and studied the similarity and difference of motivations between participants from industry and academia background. Lastly, I also explored how the functional design of KG facilitates the knowledge acquisition process and enhances the trainee's confidence level by consolidating hands-on lab-based learning with cognitive learning (concepts/knowledge in KG). All participants shared the eagerness to continue training in cybersecurity and were all interested in using our PBL lab system with KG guidance for future training.

During the development of ThoTh Lab platform, I also learnt the importance of user interface (UI) and user experience (UX) design. A good UI design will attract more learners, increase learner satisfaction and confidence, provides better learning experience, and ultimately improve students learning outcome. Without a great UI/UX design, the connection between our platform and students will be broken, and no machine learning model or personalization will work no matter how accurate the system is. Thus, a significate amount effort was spent to create the ThoTh Lab UI and improve it continuously over the years.

In the end, I'm happy to report that, based on current gathered result, ThoTh Lab is able to achieve it original designed goals. It is able to motivate students with its personalized, user-

friendly and easy-to-use graphical user interface to spend more efforts on hands-on labs and increase students' awareness of cybersecurity area. By introducing personalized learning into cybersecurity hands-on lab experience, data shows that students' experience was improved, and students expressed more interested in cybersecurity domain. ThoTh Lab is also able to improve students learning efficiency and award them with better learning outcome.

5.2 Future Work

In future work, I want to incorporate more unstructured data into our system, including but not limited to textbooks, internet web pages, and online video transcripts. In English language domain, there are several datasets that contain similar word pairs defined by human experts, including Rubenstein and Goodenough dataset [34] and WordSim353 dataset [35]. These datasets can be used as evaluation baseline for NLP processing modules in English language domain. But such dataset is absent in cybersecurity domain. As a result, we can only relay on our own domain knowledge to check the results and fine-tune model parameter base on our own judgment. However, word embedding using unsupervised learning methods like Word2Vec is still the mainstream method on natural language 41 dataset, as these datasets are way too large for human experts to supervise the learning process.

One possible solution to these challenges is constructing an ontology with a group of experts in cybersecurity. A few examples of such ontology emerge in recent research works [36] [37]. We plan to incorporate cybersecurity ontology which is intended to support the knowledge graph generation. By adding ontology in the knowledge graph, edges in the

knowledge graph will get the semantic definition which is much more useful than the similarity value currently used. The goal is to build a knowledge graph that will serve as the backbone of the cybersecurity education domain, which would evolve and grow with additional cybersecurity lab sets as they become available, and fully adaptive to different learners who want to utilize it.

The findings in this dissertation also identified the urgency of developing a more advanced and complete cybersecurity knowledge base that covers the majority of cybersecurity concepts and training scenarios. A personalized knowledge graph for an individual trainee is also required to gauge the problem-based learning experience in this system more accurately. Lastly, since this study only samples on a specific group of participants in a cybersecurity class or a professional training event, the study base is limited. Further experiments and in-class studies are necessary.

Lastly, we want to continue working on the blockchain lab content storage module by setting up the proposed system and testing its performance.

REFERENCES

[1] Amazon, E. C. "Amazon web services." Available in: http://aws.amazon.com/es/ec2/(November 2017).

[2] Jenne, Randy. "MyitLab." Proceedings of the 14th Western Canadian Conference on Computing Education. 2009.

[3] Copeland, Marshall, et al. "Microsoft Azure." New York, NY, USA:: Apress (2015).

[4] Ricci, Robert, Eric Eide, and CloudLab Team. "Introducing CloudLab: Scientific infrastructure for advancing cloud architectures and applications." the magazine of USENIX & SAGE 39.6 (2014): 36-38.

[5] Office of Educational Technology. "National Education Technology Plan." (2017).

[6] Honey, Peter, and Alan Mumford. The learning styles helper's guide. Maidenhead: Peter Honey Publications, 2000.

[7] Felder, Richard M., and Linda K. Silverman. "Learning and teaching styles in engineering education." Engineering education 78.7 (1988): 674-681.

[8] Liyanage, Madura Prabhani Pitigala, Lasith Gunawardena KS, and Masahito Hirakawa. "Detecting learning styles in learning management systems using data mining." Journal of Information Processing 24.4 (2016): 740-749.

[9] Chang, Yi-Chun, et al. "A learning style classification mechanism for e-learning." Computers & Education 53.2 (2009): 273-285.

[10] Kolekar, Sucheta V., S. G. Sanjeevi, and D. S. Bormane. "Learning style recognition using artificial neural network for adaptive user interface in e-learning." 2010 IEEE International Conference on Computational Intelligence and Computing Research. IEEE, 2010.

[11] Graf, Sabine, Kinshuk, and Tzu-Chien Liu. "Supporting teachers in identifying students' learning styles in learning management systems: An automatic student modelling approach." Journal of Educational Technology & Society 12.4 (2009): 3-14.

[12] Villaverde, Jorge Eduardo, Daniela Godoy, and Analía Amandi. "Learning styles' recognition in e-learning environments with feed-forward neural networks." Journal of Computer Assisted Learning 22.3 (2006): 197-206.

[13] Barber, Rebecca, and Mike Sharkey. "Course correction: Using analytics to predict course success." Proceedings of the 2nd international conference on learning analytics and knowledge. 2012.

[14] Myller, Niko, Jarkko Suhonen, and Erkki Sutinen. "Using data mining for improving web-based course design." International Conference on Computers in Education, 2002. Proceedings. IEEE, 2002.

[15] Kotsiantis, Sotiris B., and Panayiotis E. Pintelas. "Predicting students marks in hellenic open university." Fifth IEEE International Conference on Advanced Learning Technologies (ICALT'05). IEEE, 2005.

[16] Calvo-Flores, M. Delgado, et al. "Predicting students' marks from Moodle logs using neural network models." Current Developments in Technology-Assisted Education 1.2 (2006): 586-590.

[17] Pappano, Laura. "The Year of the MOOC." The New York Times 2.12 (2012): 2012.

[18] Käser, Tanja, et al. "Dynamic Bayesian networks for student modeling." IEEE Transactions on Learning Technologies 10.4 (2017): 450-462.

[19] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems. 2013.

[20] Fellbaum, Christiane. "WordNet." The encyclopedia of applied linguistics (2012).

[21] Bishop, Matt, et al. "Cybersecurity curricular guidelines." IFIP World Conference on Information Security Education. Springer, Cham, 2017.

[22] Newhouse, William, et al. "National initiative for cybersecurity education (NICE) cybersecurity workforce framework." NIST Special Publication 800 (2017): 181.

[23] Conklin, Wm, and Matt Bishop. "Contrasting the CSEC 2017 and the CAE Designation Requirements." (2018).

[24] Caruso, Giovanni, Lucia Ferlino, and Luigi Oliva. "WikiMindMap, uno strumento versatile per la didattica." Italian Journal of Educational Technology 18.2 (2010): 59-59.

[25] Mahdisoltani, Farzaneh, Joanna Biega, and Fabian M. Suchanek. "Yago3: A knowledge base from multilingual wikipedias." 2013.

[26] Nickel, Maximilian, et al. "A review of relational machine learning for knowledge graphs." Proceedings of the IEEE 104.1 (2015): 11-33.

[27] Milne, David, and Ian H. Witten. "Learning to link with wikipedia." Proceedings of the 17th ACM conference on Information and knowledge management. 2008.

[28] Tsai, Chen-Tse, and Dan Roth. "Cross-lingual wikification using multilingual embeddings." Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2016.

[29] Grefenstette, Gregory, and Lawrence Muchemi. "Determining the characteristic vocabulary for a specialized dictionary using word2vec and a directed crawler." arXiv preprint arXiv:1605.09564 (2016).

[30] Speer, Robyn, Joshua Chin, and Catherine Havasi. "Conceptnet 5.5: An open multilingual graph of general knowledge." Thirty-First AAAI Conference on Artificial Intelligence. 2017.

[31] Soloman, Barbara A., and Richard M. Felder. "Index of learning styles questionnaire." NC State University. Available online at: http://www. engr. ncsu. edu/learningstyles/ilsweb. html (last visited on 14.05. 2010) 70 (2005).

[32] Goldberg, Yoav, and Omer Levy. "word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method." arXiv preprint arXiv:1402.3722 (2014).

[33] Heracleous, Loizos, Julia Gößwein, and Philippe Beaudette. "Open strategy-making at the Wikimedia foundation: a dialogic perspective." The Journal of Applied Behavioral Science 54.1 (2018): 5-35.

[34] Rubenstein, Herbert, and John B. Goodenough. "Contextual correlates of synonymy." Communications of the ACM 8.10 (1965): 627-633.

[35] Finkelstein, Lev, et al. "Placing search in context: The concept revisited." Proceedings of the 10th international conference on World Wide Web. 2001.

[36] Obrst, Leo, Penny Chase, and Richard Markeloff. "Developing an Ontology of the Cyber Security Domain." STIDS. 2012.

[37] Oltramari, Alessandro, et al. "Building an Ontology of Cyber Security." STIDS. 2014.

[38] Böhme, Rainer, et al. "Bitcoin: Economics, technology, and governance." Journal of economic Perspectives 29.2 (2015): 213-38.

[39] Pongnumkul, Suporn, Chaiyaphum Siripanpornchana, and Suttipong Thajchayapong. "Performance analysis of private blockchain platforms in varying workloads." 2017 26th International Conference on Computer Communication and Networks (ICCCN). IEEE, 2017.

[40] Benet, Juan. "Ipfs-content addressed, versioned, p2p file system." arXiv preprint arXiv:1407.3561 (2014).

[41] Chine, Karim. "Learning math and statistics on the cloud, towards an EC2-based Google Docs-like portal for teaching/learning collaboratively with R and Scilab." 2010 10th IEEE International Conference on Advanced Learning Technologies. IEEE, 2010.

[42] González-Martínez, José A., et al. "Cloud computing and education: A state-of-the-art survey." Computers & Education 80 (2015): 132-151.

[43] Maldarelli, Grace A., et al. "Virtual lab demonstrations improve students' mastery of basic biology laboratory techniques." Journal of Microbiology & Biology Education 10.1 (2009): 51-57.

[44] Makransky, Guido, Malene Warming Thisgaard, and Helen Gadegaard. "Virtual simulations as preparation for lab exercises: Assessing learning of key laboratory skills in microbiology and improvement of essential non-cognitive skills." PloS one 11.6 (2016): e0155895.

[45] Darrah, Marjorie, et al. "Are virtual labs as effective as hands-on labs for undergraduate physics? A comparative study at two major universities." Journal of science education and technology 23.6 (2014): 803-814.

[46] Tatli, Zeynep, and Alipaşa Ayas. "Virtual laboratory applications in chemistry education." Procedia-Social and behavioral sciences 9 (2010): 938-942.

[47] Deng, Yuli, et al. "Personalized learning in a virtual hands-on lab platform for computer science education." 2018 IEEE Frontiers in Education Conference (FIE). IEEE, 2018.

[48] Deng, Yuli, Dijiang Huang, and Chun-Jen Chung. "ThoTh Lab: A personalized learning framework for CS hands-on projects." Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education. 2017.

[49] Syamsuddin, Irfan. "VILARITY-Virtual Laboratory for Information Security Practices." TEM Journal 8.3 (2019): 1011-1016.

[50] Alharbi, Ali H. "A Portable Virtual LAB for Informatics Education using Open Source Software." Int. J. Adv. Comput. Sci. Appl 9.2 (2018).

[51] Yang, Wenli, et al. "A distributed case-and project-based learning to design 3D lab on electronic engineering education." Computer Applications in Engineering Education 27.2 (2019): 430-451.

[52] Santos, Gilberto, et al. "Engineering learning objectives and computer assisted tools." European Journal of Engineering Education 44.4 (2019): 616-628.

[53] Yi, Zhou, Jiang Jian-Jun, and Fan Shao-Chun. "A LabVIEW-based, interactive virtual laboratory for electronic engineering education." International Journal of Engineering Education 21.1 (2005): 94-102.

[54] Kapilan, Natesan, P. Vidhya, and Xiao-Zhi Gao. "Virtual laboratory: A boon to the mechanical engineering education during covid-19 pandemic." Higher Education for the Future 8.1 (2021): 31-46.

[55] Salmerón-Manzano, Esther, and Francisco Manzano-Agugliaro. "The higher education sustainability through virtual laboratories: The Spanish University as case of study." Sustainability 10.11 (2018): 4040.

[56] Muradova, Firuza Rashidovna. "VIRTUAL LABORATORIES IN TEACHING AND EDUCATION." Theoretical & Applied Science 2 (2020): 106-109.

[57] Grout, Ian. "Remote laboratories as a means to widen participation in STEM education." Education Sciences 7.4 (2017): 85.

[58] Stark, Erich, et al. "Virtual laboratory based on Node. js technology." 2017 21st International Conference on Process Control (PC). IEEE, 2017.

[59] Ghayoor, F. "A MATLAB-based virtual robotics laboratory: Demonstrated by a two-wheeled inverted pendulum." The International Journal of Electrical Engineering & Education 57.4 (2020): 301-320.

[60] Kumar, Dhanush, et al. Virtual and remote laboratories augment self learning and interactions: Development, deployment and assessments with direct and online feedback. No. e26715v1. PeerJ Preprints, 2018.

[61] Perales, Mikel, Luis Pedraza, and Pablo Moreno-Ger. "Work-in-progress: Improving online higher education with virtual and remote labs." 2019 IEEE Global Engineering Education Conference (EDUCON). IEEE, 2019.

[62] Barrows, Howard S. "Problem-based learning in medicine and beyond: A brief overview." New directions for teaching and learning 1996.68 (1996): 3-12.

[63] Driessen, Erik, and Cees Van Der Vleuten. "Matching student assessment to problem-based learning: lessons from experience in a law faculty." Studies in Continuing Education 22.2 (2000): 235-248.

[64] Stinson, John E., and Richard G. Milter. "Problem-based learning in business education: Curriculum design and implementation issues." New directions for teaching and learning 1996.68 (1996): 33-42.

[65] Schmidt, Henk G., et al. "Problem-based learning is compatible with human cognitive architecture: Commentary on Kirschner, Sweller, and." Educational psychologist 42.2 (2007): 91-97.

[66] Mills, Julie E., and David F. Treagust. "Engineering education—Is problem-based or project-based learning the answer." Australasian journal of engineering education 3.2 (2003): 2-16.

[67] Perrenet, Jacob C., Peter AJ Bouhuijs, and Jan GMM Smits. "The suitability of problem-based learning for engineering education: theory and practice." Teaching in higher education 5.3 (2000): 345-358.

[68] Yadav, Aman, et al. "Problem-based learning: Influence on students' learning in an electrical engineering course." Journal of Engineering Education 100.2 (2011): 253-280.

[69] Dumais, Susan T. "Latent semantic analysis." Annual review of information science and technology 38.1 (2004): 188-230.

[70] Paige, Christopher C., and Michael A. Saunders. "Towards a generalized singular value decomposition." SIAM Journal on Numerical Analysis 18.3 (1981): 398-405.

[71] Raj, Rajendra K., and Allen Parrish. "Toward standards in undergraduate cybersecurity education in 2018." Computer 51.2 (2018): 72-75.

[72] Woodward, Belle, Thomas Imboden, and Nancy L. Martin. "An undergraduate information security program: More than a curriculum." Journal of Information Systems Education 24.1 (2013): 63.

[73] Endicott-Popovsky, Barbara E., and Viatcheslav M. Popovsky. "Application of pedagogical fundamentals for the holistic development of cybersecurity professionals." ACM Inroads 5.1 (2014): 57-68.

[74] Knapp, Kenneth J., Christopher Maurer, and Miloslava Plachkinova. "Maintaining a cybersecurity curriculum: Professional certifications as valuable guidance." Journal of Information Systems Education 28.2 (2017): 101.

[75] Said, Samuel Essa. Pedagogical Best Practices in Higher Education National Centers of Academic Excellence/Cyber Defense Centers of Academic Excellence in Cyber Defense. Diss. Union University, 2018.

[76] Newhouse, William, et al. "National initiative for cybersecurity education (NICE) cybersecurity workforce framework." NIST special publication 800.2017 (2017): 181.

[77] Burley, Diana, et al. "ACM Joint Task Force on Cybersecurity Education." Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education. 2017.

[78] Deng, Yuli, et al. "Knowledge graph-based learning guidance for cybersecurity hands-on labs." Proceedings of the ACM conference on global computing education. 2019.

[79] Shi, Daqian, et al. "A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning." Knowledge-Based Systems 195 (2020): 105618.

[80] Wang, Quan, et al. "Knowledge graph embedding: A survey of approaches and applications." IEEE Transactions on Knowledge and Data Engineering 29.12 (2017): 2724-2743.

[81] Ramos, Juan. "Using tf-idf to determine word relevance in document queries." Proceedings of the first instructional conference on machine learning. Vol. 242. No. 1. 2003.

[82] Du, Wenliang. "SEED: hands-on lab exercises for computer security education." IEEE Security & Privacy 9.5 (2011): 70-73.

[83] Vieira, Susana M., Uzay Kaymak, and João MC Sousa. "Cohen's kappa coefficient as a performance measure for feature selection." International Conference on Fuzzy Systems. IEEE, 2010.

[84] Keller, J. Motivational design of instruction. Instructional design theories and models. Reigeluth, C. Diss. ed.), S. 386 ff.

[85] Loorbach, Nicole, et al. "Validation of the Instructional Materials Motivation Survey (IMMS) in a self-directed instructional setting aimed at working with technology." British journal of educational technology 46.1 (2015): 204-218.

[86] Cook, David A., et al. "Measuring motivational characteristics of courses: applying Keller's instructional materials motivation survey to a web-based course." Academic Medicine 84.11 (2009): 1505-1509.

[87] Shah, Kartik, et al. "A Qualitative Study on How Students Interact with Quizzes and Estimate Confidence on Their Answers." Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education V. 1. 2021.

[88] Kinnunen, Päivi, and Beth Simon. "Building theory about computing education phenomena: a discussion of grounded theory." Proceedings of the 10th Koli Calling International Conference on Computing Education Research. 2010.

[89] Adams, Dennis A., R. Ryan Nelson, and Peter A. Todd. "Perceived usefulness, ease of use, and usage of information technology: A replication." MIS quarterly (1992): 227-247.

[90] Dai, Jun. "Situation awareness-oriented cybersecurity education." 2018 IEEE Frontiers in Education Conference (FIE). IEEE, 2018.

[91] Gerstner, Sabine, and Franz X. Bogner. "Cognitive Achievement and Motivation in Hands-on and Teacher-Centred Science Classes: Does an additional hands-on consolidation phase (concept mapping) optimise cognitive learning at work stations?" International Journal of Science Education 32.7 (2010): 849-870.

[92] Goodhue, Dale L., and Ronald L. Thompson. "Task-technology fit and individual performance." MIS quarterly (1995): 213-236.

[93] McGill, Tanya J., and Jane E. Klobas. "A task–technology fit view of learning management system impact." Computers & Education 52.2 (2009): 496-508.

[94] Bernard, Jason, et al. "Learning style Identifier: Improving the precision of learning style identification through computational intelligence algorithms." Expert Systems with Applications 75 (2017): 94-108.

[95] Graf, Sabine, Kinshuk, and Tzu-Chien Liu. "Supporting teachers in identifying students' learning styles in learning management systems: An automatic student modelling approach." Journal of Educational Technology & Society 12.4 (2009): 3-14.

[96] García, Patricio, et al. "Evaluating Bayesian networks' precision for detecting students' learning styles." Computers & Education 49.3 (2007): 794-808.

[97] Özpolat, Ebru, and Gözde B. Akar. "Automatic detection of learning styles for an e-learning system." Computers & Education 53.2 (2009): 355-367.

APPENDIX A

INDEX OF LEARNING STYLES (ILS) LEARNING STYLE QUESTIONNAIRE

This questionnaire is designed to find out what you learning preferences are. It was originally designed by Felder and Silverman at North Carolina State University, USA.

To complete the questionnaire please circle "a" or "b" to indicate your answer to every question. You may only choose one answer for each question, and you must answer every question. If both "a" and "b" seem to apply to you, please choose the one that applies more frequently.

1. I understand something better after I

(a) try it out.

(b) think it through.

2. I would rather be considered

(a) realistic.

(b) innovative.

3. When I think about what I did yesterday, I am most likely to get

(a) a picture.

(b) words.

4. I tend to

(a) understand details of a subject but may be fuzzy about its overall structure.

(b) understand the overall structure but may be fuzzy about details.

5. When I am learning something new, it helps me to

(a) talk about it.

(b) think about it.

6. If I were a teacher, I would rather teach a course

(a) that deals with facts and real-life situations.

(b) that deals with ideas and theories.

7. I prefer to get new information in

(a) pictures, diagrams, graphs, or maps.

(b) written directions or verbal information.

8. Once I understand

(a) all the parts, I understand the whole thing.

(b) the whole thing, I see how the parts fit.

9. In a study group working on difficult material, I am more likely to

(a) jump in and contribute ideas.

(b) sit back and listen.

10. I find it easier

(a) to learn facts.

(b) to learn concepts.

11. In a book with lots of pictures and charts, I am likely to

(a) look over the pictures and charts carefully.

(b) focus on the written text.

12. When I solve math problems

(a) I usually work my way to the solutions one step at a time.

(b) I often just see the solutions but then have to struggle to figure out the steps to get to them.

13. In classes I have taken

(a) I have usually got to know many of the students.

(b) I have rarely got to know many of the students.

14. In reading non-fiction, I prefer

(a) something that teaches me new facts or tells me how to do something.

(b) something that gives me new ideas to think about.

15. I like teachers

(a) who put a lot of diagrams on the board.

(b) who spend a lot of time explaining.

16. When I'm analyzing a story or a novel

(a) I think of the incidents and try to put them together to figure out the themes.

(b) I just know what the themes are when I finish reading and then I have to go back and find the incidents that demonstrate them.

17. When I start a homework problem, I am more likely to

(a) start working on the solution immediately.

(b) try to fully understand the problem first.

18. I prefer the idea of

(a) certainty.

(b) theory.

19. I remember best

(a) what I see.

(b) what I hear.

20. It is more important to me that an instructor

(a) lay out the material in clear sequential steps.

(b) give me an overall picture and relate the material to other subjects.

21. I prefer to study

(a) in a group.

(b) alone.

22. I am more likely to be considered

(a) careful about the details of my work.

(b) creative about how to do my work.

23. When I get directions to a new place, I prefer

(a) a map.

(b) written instructions.

24. I learn

(a) at a fairly regular pace. If I study hard, I'll "get it."

(b) in fits and starts. I'll be totally confused and then suddenly it all "clicks."

25. I would rather first

(a) try things out.

(b) think about how I'm going to do it.

26. When I am reading for enjoyment, I like writers to

(a) clearly say what they mean.

(b) say things in creative, interesting ways.

27. When I see a diagram or sketch in class, I am most likely to remember

(a) the picture.

(b) what the instructor said about it.

28. When considering a body of information, I am more likely to

(a) focus on details and miss the big picture.

(b) try to understand the big picture before getting into the details.

29. I more easily remember

(a) something I have done.

(b) something I have thought a lot about.

30. When I have to perform a task, I prefer to

(a) master one way of doing it.

(b) come up with new ways of doing it.

31. When someone is showing me data, I prefer

(a) charts or graphs.

(b) text summarizing the results.

32. When writing a paper, I am more likely to

(a) work on (think about or write) the beginning of the paper and progress forward.

(b) work on (think about or write) different parts of the paper and then order them.

33. When I have to work on a group project, I first want to

(a) have a "group brainstorming" where everyone contributes ideas.

(b) brainstorm individually and then come together as a group to compare ideas.

34. I consider it higher praise to call someone

(a) sensible.

(b) imaginative.

35. When I meet people at a party, I am more likely to remember

(a) what they looked like.

(b) what they said about themselves.

36. When I am learning a new subject, I prefer to

(a) stay focused on that subject, learning as much about it as I can.

(b) try to make connections between those subject and related subjects.

37. I am more likely to be considered

(a) outgoing.

(b) reserved.

38. I prefer courses that emphasize

(a) concrete material (facts, data).

(b) abstract material (concepts, theories).

39. For entertainment, I would rather

(a) watch television.

(b) read a book.

40. Some teachers start their lectures with an outline of what they will cover. Such outlines are

(a) somewhat helpful to me.

(b) very helpful to me.

41. The idea of doing homework in groups, with one grade for the entire group,

(a) appeals to me.

(b) does not appeal to me.

42. When I am doing long calculations,

(a) I tend to repeat all my steps and check my work carefully.

(b) I find checking my work tiresome and have to force myself to do it.

43. I tend to picture places I have been

(a) easily and fairly accurately.

(b) with difficulty and without much detail.

44. When solving problems in a group, I would be more likely to

(a) think of the steps in the solution process.

(b) think of possible consequences or applications of the solution in a wide range of areas.

APPENDIX B

LEARNING STYLES QUESTIONNAIRE SCORING SHEET

Place a “1” in the appropriate spaces in the table below (e.g., if you answered "a" to Question 3, put a "1" in Column "a" by Question 3).

Add up the columns and write the totals in the indicated spaces.

For each of the four scales, subtract the smaller total from the larger one. Write the difference (1 to 11) and the letter (a or b) with the larger total.

Activist/Reflector			**Sensing/Intuitive**			**Visual/Verbal**			**Sequential/Global**		
Q	*a*	*b*	Q	*a*	*b*	*Q*	*a*	*b*	*Q*	*a*	*b*
1			*2*			*3*			*4*		
5			*6*			*7*			*8*		
9			*10*			*11*			*12*		
13			*14*			*15*			*16*		
17			*18*			*19*			*20*		
21			*22*			*23*			*24*		
25			*26*			*27*			*28*		
29			*30*			*31*			*32*		
33			*34*			*35*			*36*		
37			*38*			*39*			*40*		
41			*42*			*43*			*44*		
Total (add up each column)											
Activist/Reflector			**Sensing/Intuitive**			**Visual/Verbal**			**Sequential/Global**		
Q	a	b	Q	a	b	Q	a	b	Q	a	b
Larger – Smaller + Letter of Larger (see below)*											

<u>Example</u>: If your total was 3 for a and 8 for b: 8 – 3 = 5, b is letter of larger so you would enter 5b.

95

APPENDIX C

THOTH LAB STUDENT EXIT SURVEY VER.A

Q1: The virtual lab platform is convenient to access.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q2: Doing labs in virtual lab platform is easier compared to doing labs in a physical lab.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q3: Personal knowledge graph in the virtual lab platform is accurate at the beginning of the class.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q4: Personal knowledge graph is accurate at the end of the class.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q5: I regularly check my personal knowledge graph.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q6: Extra questions for research volunteers for the recommendation system: Q6: The recommendation a reasonable recommendation for me.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q7: The connection/similarity between labs recommended to me is noticeable.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q8: The recommendation system is easy to use.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q9: Compare to labs required by the course, I find the labs recommended to me more interesting.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

Q10: I want to keep on using the system as a self-guidance tool after this class.

a. Totally disagree

b. Disagree

c. Neutral

d. Agree

e. Fully agree

APPENDIX D

THOTH LAB STUDENT EXIT SURVEY VER.B

General

Created based on Instructional Materials Motivation Survey to identify student motivation when doing the lab

General

1. Have you been (motivated to) learn computer science security with a project-based learning approach?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

2. Do you think that the project-based learning approach has influenced your learning?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

3. Do you consider the labs we did in this class close to real world?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

4. Do you consider these projects important for your own professional growth?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

Attention

5. The lab martial and lab platform helped to hold my attention

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

6. The way the information is arranged in the lab instructions and lab platform helped keep my attention

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

7. The variety of reading materials, exercises, illustrations, etc., helped keep my attention on the lab.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

Relevance

8. It is clear to me how the content of these lab materials is related to things l learn during class videos and slides.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

9. The content in the lab is relevant to my interests and worth knowing

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

10. The content of these lab will be useful to me in the future

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

Project-relationship

11. Project 1 is necessary for me, as it prepared me well for Project 2,3 and 4?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

12. It is clear to me that Project 2 and Project 3 are more related when compared to Project 1 and 4.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

13. Instructor should keep Project 2 and 3 separates, proceeding in an orderly way and step by step, instead of merge Project 2 and 3 together.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

14. Project 4 is closely related to other Projects.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

Confidence

15. As I worked with these lab materials, I was confident that I could learn computer network security well

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

16. After reading these lab instructions, I was confident that I would be able to complete labs and the class well.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

17. I could not really understand quite a bit of the material in the lab instruction.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

Satisfaction

18. I enjoyed working with these labs so much that I was stimulated to keep learning more about network security and other related topics.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

19. It felt good to successfully accomplish lab tasks.

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

20. The feedback from instructor helped me feel rewarded for my efforts in doing the lab

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

21. Do you feel satisfied with the results delivered by you?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

Role-based

22. Do you think that including a “attacker” role in the project-based learning approach would benefit you in view of a future real professional situation?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

23. Do you consider the use of role playing (attacker/defender/victim) important?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

Project-based

24. Do you believe that the use of the project-based learning approach has helped you to develop your learning skills?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

25. Do you consider significant the extra time you have devoted on the project assignments?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

26. Do you think that devoting the project's time to traditional lectures would be better?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

27. Have you enjoyed the project experience?

Strongly disagree Disagree Neutral Agree Strongly agree No opinion

Do you have any additional comment, critics or suggestion regarding the project setup, running, etc.?

Do you have any additional comment, critics or suggestion regarding the ThoTh Lab Platform?

APPENDIX E

UNIVERSITY HUMAN SUBJECTS INSTITUTIONAL REVIEW BOARD (IRB)

APPROVAL DOCUMENT

EXEMPTION GRANTED

Dijiang Huang

CIDSE: Computing, Informatics and Decision Systems Engineering, School of
480/965-2776

Dijiang.Huang@asu.edu

Dear Dijiang Huang:

On 12/29/2020 the ASU IRB reviewed the following protocol:

Type of Review:	Initial Study
Title:	AISecKG: AI for Cybersecurity Education via an ML-enabled Security Knowledge Graph
Investigator:	Dijiang Huang
IRB ID:	STUDY00013036
Funding:	Name: National Science Foundation (NSF), GrantOffice ID: FP27392
Grant Title:	FP27392;
Grant ID:	FP27392;
Documents Reviewed:	• AISecKG, Category: Sponsor Attachment; • Consent Form, Category: Consent Form; • IRB data collection protocol, Category: IRB Protocol; • supporting documents 29-12-2020, Category: Other; • Survey Questions for students and instructors, Category: Measures (Survey questions/Interview questions /interview guides/focus group questions);

The IRB determined that the protocol is considered exempt pursuant to Federal Regulations 45CFR46 (1) Educational settings, (2) Tests, surveys, interviews, or observation on 12/29/2020.

In conducting this protocol you are required to follow the requirements listed in the INVESTIGATOR MANUAL (HRP-103).

If any changes are made to the study, the IRB must be notified at research.integrity@asu.edu to determine if additional reviews/approvals are required.Changes may include but not limited to revisions to data collection, survey and/or interview questions, and vulnerable populations, etc.

Sincerely,

IRB

Administrator

109