A Hybrid Approach to Developing a Cyber Security Ontology

James Geller[1], Soon Ae Chun[2] and Arwa Wali[1,3]
[1]*New Jersey Institute of Technology, Newark, NJ, U.S.A.*
[2]*City University of New York – College of Staten Island, Staten Island, NY, U.S.A.*
[3]*College of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia*

Keywords: Cyber Security Ontology; Security Knowledge Acquisition Tool, Security Learning Objects, Textbook, Index Terms, Augmented Ontology.

Abstract: The process of developing an ontology cannot be fully automated at the current state-of-the-art. However, leaving the tedious, time-consuming and error-prone task of ontology development entirely to humans has problems of its own, including limited staff budgets and semantic disagreements between experts. Thus, a hybrid computer/expert approach is advocated. The research challenge is how to minimize and optimally organize the task of the expert(s) while maximally leveraging the power of the computer and of existing computer-readable documents. The purpose of this paper is two-fold. First we present such a hybrid approach by describing a knowledge acquisition tool that we have developed. This tool makes use of an existing Bootstrap Ontology and proposes likely locations of concepts and semantic relationships, based on a text book, to a domain expert who can decide on them. The tool is attempting to minimize the number of interactions. Secondly we are proposing the notion of an augmented ontology specifically for pedagogical use. The application domain of this work is cyber-security education, but the ontology development methods are applicable to any educational topic.

1 INTRODUCTION

Building ontologies is difficult. Two major schools of thought with various combinations have defined the state-of-the-art. One school relies entirely on human experts. This approach works well as long as the desired ontology is small. However, most useful ontologies tend to be large. For example, in Medical Informatics, SNOMED CT (Cornet & de Keizer, 2008) contains about 400,000 concepts. Due to differing viewpoints, large ontologies cannot be built in a completely modular fashion. Rather, experts working on different modules need to communicate with each other, which increases the amount of time that is necessary for building the final product.

The alternative approach to ontology building is to automate the process by letting "a computer read available material, just like a human would read it." Unfortunately, this approach can only become completely successful after the Turing Test has been passed. This leads us to a two-pronged approach. (1) Use any relevant structured or semi-structured information that is available besides free text. (2) Implement a software tool that maximizes support for human experts and minimizes the time they need to spend on ontology building.

An et al. and Geller et al. (An, Geller, Wu, & Chun, 2007; Geller, Chun, & An, 2008) have used the Deep Web as a source of structured information. However, web site owners are increasingly unwilling to let robot programs extract backend data, which makes this approach difficult. Thus, we have turned to another source of domain knowledge.

The best Knowledge Representation "system" for over 3000 years has been the book. A major improvement was achieved by adding a table of contents with page numbers and a back index to books. The exact history of the "alphabetic index" is difficult to trace, but (Cleveland & Cleveland, 2013) mentions an example from the 5th century. The fact that a back-of-the-book index incorporates "a degree of intelligence" can be seen from Artificial Intelligence attempts to automate the process of building an index (Wu, Li, Mitra, & Giles, 2013).

Our approach is to make use of the intelligence that went into building an index and of the structure of the index as it is intertwined with the text in the body of the book. In this paper, we are describing a hybrid approach to developing a cyber-security ontology for education. First a seed ontology is

Geller J., Ae Chun S. and Wali A..
A Hybrid Approach to Developing a Cyber Security Ontology.
DOI: 10.5220/0005111503770384
In *Proceedings of 3rd International Conference on Data Management Technologies and Applications* (DATA-2014), pages 377-384
ISBN: 978-989-758-035-2

extended in a semi-automatic fashion, resulting in the Bootstrap Ontology. The Bootstrap Ontology is then extended by a domain expert using a knowledge acquisition tool, called the Security Knowledge Acquisition Tool (SKAT) that was developed in this research. The semi-automatic ontology construction component uses the textbook index terms as input and classifies each index term into a seed ontology of concepts. (Note that when we use the term "classify," this is *not* the DL classification algorithm.) This preliminary work was reported in (Chun, Geller, & Wali, 2014; Wali, Chun, & Geller, 2013). However, this semi-automatic processing left many index terms unclassified in the security ontology.

The SKAT Tool allows a security domain expert to manually place the security terms into the ontology. In order to alleviate the cognitive burden on the domain expert, the tool parses the textbook to identify index terms co-occurring with ontology concepts to make suggestions of candidate concepts and relationships. Thus, SKAT incorporates a *Concept Recommendation component* to identify occurrences of ontology concepts in the unstructured text of a textbook that co-occur with the semi-structured index terms, and it recommends these index terms as candidate concepts to the SKAT user. This will minimize the expert's effort to search in the ontology structure for concepts that may be related to a security term from the text book.

In this paper, we present related work in Section 2, and briefly review our automatic concept classification approach and its results in Section 3. Section 4 presents our approach to augmenting the cyber security ontology further through the use of concept recommendation and the SKAT tool, as utilized by a human expert. In Section 5, we discuss the results and experience with the SKAT tool. Section 6 concludes with a summary and future research tasks.

2 RELATED WORK

2.1 Security Ontologies

Ontology development approaches can be divided into the purely manual, the purely automatic, and various hybrid methods. In the manual approach, a team of human experts accumulates domain concepts, organizes them into a subconcept (subclass or IS-A) hierarchy and then includes additional information, such as semantic relationships between pairs of concepts or details about concepts.

LoLaLi is an example of an ontology that was manually built (Caracciolo, 2006). Building LoLaLi manually was very time consuming.

The automatic methods attempt to build an ontology by parsing of English text. There are several variants for this approach, such as clustering, linguistic pattern matching, formal concept analysis, or ontology alignment. Hindle's work is based on the clustering approach (Hindle, 1990).

Hearst et al. used a linguistic pattern matching approach to find semantic relationships between terms from large corpora (Hearst, 1992). Formal concept analysis was used for extracting monotonic inheritance relations from unstructured data (Cimiano, Hotho, & Staab, 2005; Wiebke, 2004).

Another method for developing a comprehensive ontology is by ontology alignment. BLOOMS is an example for building an ontology by alignment or ontology matching from smaller ontologies (Jain, Hitzler, Sheth, Verma, & Yeh, 2010).

In previous research, we used a methodology for automatic construction of a domain ontology, by combining WordNet (Fellbaum, 1998) concepts with domain-specific concept information extracted from the web (An et al., 2007). Methods based on unstructured data from the web suffer from web pages that may be changing rapidly. Pattanasri et al. (Pattanasri, Jatowt, & Tanaka, 2007) developed a textbook ontology using the index and the table of contents. The concepts in each ontology are cross-referenced with page numbers to refer to corresponding textbook segments or slide page numbers. To the best of our knowledge, our research is the first that combines a back of the textbook index with an existing security ontology as a seed structure to build a more complete Bootstrap Ontology of cyber-security terms and then provides a tool for a human expert to augment the Bootstrap Ontology. Several preexisting security-related ontologies have been reported (Bajec, Eder, Souag, Salinesi, & Comyn-Wattiau, 2012; Fenz & Ekelhart, 2009; Geneiatakis & Lambrinoudakis, 2007; Herzog, Shahmeri, & Duma, 2007; Meersman, Tari, Kim, Luo, & Kang, 2005; Vigna et al., 2003). For a review see (Blanco et al., 2008).

2.2 Project Environment

The work presented in this paper is part of an NSF funded project for creating an "Ultimate Course Search" (UCS) tool for students to navigate existing teaching materials. The UCS tool allows students to search video recordings of a whole semester of classes, interlinked with the power point slides used

in the lectures and page images of the text book. The cyber-security ontology is visible in the UCS tool.

The intended function of the ontology is as follows. If a search term does not appear anywhere in the power point presentation but does appear in the ontology, then the ontology component can suggest closely related search terms, such as parents or siblings, to be located in the power point presentation. The likelihood that a student will search for a term that is not correct is high, because a learning student does not know the appropriate terms of the domain. Thus, the ontology is needed.

3 AUTOMATED SECURITY ONTOLOGY DEVELOPMENT

Our previous work (Wali et al., 2013) described the bootstrapping approach to enrich a seed ontology using an ensemble of different algorithms to classify book index terms into the seed ontology. The bootstrapping approach starts with the ontology of Herzog (Herzog et al., 2007) as seed ontology. Terms from the book index of Goodrich (Goodrich & Tamassia, 2010) are extracted and classified under the existing classes in this seed ontology by assembling different matching algorithms and evidence boosting algorithms using different sources.

We started with exact matching, followed by matching using a stemmer and incorporating subterms recognizable in the index by indentation. For example, "vulnerabilities" in the textbook index matches with the concept "vulnerability" after applying the stemmer. In the next step we used substring matching together with Wikipedia categories to place index terms into the seed ontology. For instance, "replay attacks" overlaps with the Wikipedia category "Cryptographic Attacks" under "attacks," thus the system concludes that the index term defines a subcategory of "attacks."

Subsequently section headings and subsection headings as well as linguistic heuristics (e.g., in a noun-noun phrase the second noun often indicates a superclass), etc. are used. For instance, "cryptographic compression function" belongs to "Cryptography" as a section heading and as a security class name. Next, prefix and postfix modifier matching is used, e.g., to determine that "E-mail Worm" potentially belongs to "E-mail" and "Worm" as super classes (with "Worm" more likely).

In addition, NIST's security term definitions were extracted and included in the Bootstrap Ontology to define its concepts. However, a sizable number of index terms remained unclassified. Among 724 index terms, 263 terms were successfully classified into the seed ontology, which corresponds to 36.32%.

In this paper, we are approaching the problems of including concepts and incorporating semantic relationships between the remaining unclassified index terms and existing concepts in the Bootstrap Ontology. It is important to note that by including an index term in the ontology with an IS-A link, *the term is promoted to a concept*. Thus, we can talk about relationships between pairs of concepts. While including semantic relationships, we confront the following problem.

Assuming that there are N concepts in an ontology, the formula for the number of distinct pairs of concepts is $(N^2 - N)/2$. For a moderately sized ontology of 1000 concepts, that would mean 499,500 possible pairs. If a domain ontology allows for the use of IS-A relationships and nine other semantic relationships between pairs of concepts, an expert would need to consider each one of them for every pair of concepts. In reality, ten is a gross underestimate, and the expert needs to also entertain the possibility that no relationship at all holds.

If an expert could make a decision about each pair in ten seconds, he would need 173 work days of 8 hours to review all pairs. Resources at this level are rarely available, as experts are normally busy in their field of expertise. Thus, the task of assigning semantic relationships must be minimized as much as possible by presenting only pairs to the expert that are highly likely to have a useful relationship.

4 SKAT: EXPERT KNOWLEDGE ACQUISITION TOOL

4.1 Basic Assumptions

Our work is based on the following four heuristics.

H1. If a word or a multi-word term appears in the index of a book, then it describes an important domain concept for the ontology of this domain.
H2. If two index terms appear close to each other in the body of a book, then it is likely that there is a subclass (IS-A) relationship or a semantic relationship between them.
H3. If two index terms appear repeatedly close to each other, then the likelihood of a relationship between them is higher and/or the importance of the relationship is higher than for one appearance.

H4. In a well written textbook, sentences are semantics-infused units to a higher degree than "k-word neighborhoods." Thus, the basic unit of being "close to each other" will be the sentence, as opposed to the neighborhood. Of course, other possible units also exist, such as paragraphs.

4.2 Formal Definition of the SKAT Knowledge Structure

The knowledge structure that is the backbone of SKAT will be formally introduced in this section.

Definition 1: A sentence S is a grammatically correct English sentence terminated by a period.

Definition 2: A book image B of a book is the set of all sentences derived from the text of a book by removing front matter, back matter, figures, tables, captions, page headers, page footers, footnotes and all levels of chapter and section headers.

An index is an alphabetical list of domain terms used in a textbook, together with the numbers of pages where the terms appear, possibly with a multi-level structure, synonyms, abbreviations, etc. However, an index will be viewed as a list of terms in this section. The use of the other elements of an index (page numbers, etc.) will be explicated later in this paper.

Definition 3: An index I is a flat list of unique index terms T_i.

$$I = <T_1, T_2, \ldots T_m> \quad (1)$$

As noted before, we assume the existence of a Bootstrap Ontology created by hand or derived from a seed ontology, as described in Section 3.

Definition 4: The concept list C of a Bootstrap Ontology consists of a one-level list of all the concepts C_i of the ontology.

$$C = <C_1, C_2, \ldots C_r> \quad (2)$$

Furthermore, a concept in C is represented by its preferred English term, as opposed to an ID, unlike in the UMLS (Humphreys & Lindberg, 1993).

Definition 5: A term-concept pair TCP_{ij} is an ordered pair that consists of a term T_i from I (but that is not in C) and a concept C_j from C such that there exists a complete sentence S in B that contains both T_i and C_j. identifiable among the words of the sentence.

$$TCP_{ij} = <T_i, C_j> \in S \ \& \ S \in B \quad (3)$$

Definition 6: The ranked list RL of term-concept pairs is the list of all TCPs

$$RL = <<T_i, C_j>\ldots<T_i, C_k> \ldots <T_m, C_n>\ldots<T_m, C_p>> \quad (4)$$

such that if $<T_i, C_j>$ appears in RL before $<T_m, C_p>$ then the frequency of T_i within pairs occurring within sentences is greater than or equal to the frequency of T_m.

Using f() as the function that returns the frequency of terms or concepts mentioned in RL and the symbol « to indicate "to the left in the list RL" then this would be expressed as

$$<T_i, C_i> \text{ « } <T_m, C_n> \leftrightarrow f(T_i) \geq f(T_m) \quad (5)$$

Furthermore, if the frequency of T_i=T_m then the same condition holds for the C_i. In other words, if $<T_i, C_j>$ appears before $<T_i, C_k>$ then the frequency of C_j within pairs occurring within sentences is greater than or equal to the frequency of C_k. We reemphasize that f() is based on the frequencies in B, not on RL, because pairs in RL are unique.

$$<T_i, C_j> \text{ « } <T_i, C_k> \leftrightarrow f(C_j) \geq f(C_k) \quad (6)$$

Definition 7: The projected term list PT consists of all unique terms in the order imposed by RL.

$$PT = <T_1, \ldots T_i, T_j, \ldots T_l> \quad (7)$$

such that T_i « T_j in RL for all i and j.

Definition 8: The projected concept list of a term T, PC(T), consists of all unique concepts in the order imposed by RL that appear with T in a term concept pair.

$$PC(T) = <C_1, \ldots C_i, C_j, \ldots C_l> \quad (8)$$

such that $<T, C_i>$ « $<T, C_j>$ for all i and j and all C_x appearing in pairs $<T, C_x>$ are included in PC(T).

The projected term list and projected concept list will allow us to precisely describe the information that is presented by SKAT to a human expert.

The formalism, as explained above, makes the optimistic assumption that there will be a TCP available in B for every term taken from I. The reality is that this is not the case. How can a term from the index that appears not connected to "anything" be linked to the ontology? We propose the use of a transitive linkage mechanism. Thus, we also allow term-term pairs.

Definition 9: A term-term pair TT is an ordered pair that consists of two terms T_i and T_j, i.e., TT = $<T_i, T_j>$.

Definition 10: A transitively liked term T_i is a term such that

$$<T_i, T_j> \in S_1 \ \& \ S_1 \in B \text{ and also } <T_i, C_k> \in S_2 \ \& \ S_2 \in B \quad (9)$$

As a result of discovering a transitively linked term, a new term concept pair

$$<T_i, C_k> \quad (10)$$

can be created from $<T_i, T_j>$ and $<T_j, C_k>$ and appended to RL. Note that the frequency of T_i in RL is by definition equal to 0, because T_i never occurs in a sentence of the textbook together with C_k. Therefore it is correct to include the new term concept pair at the end of RL. However, if several term-concept pairs with the same transitively linked term exist, they need to be ordered according to the concept frequency in those pairs.

4.3 Augmented Ontologies

In this section, we will discuss the necessary background for understanding the "importance," "difficulty," and "prerequisite" mechanisms of the SKAT tool. According to Noy and McGuiness (Noy & McGuinness), building an ontology consists of the following steps, whereby we adapt the description of the third step and omit the irrelevant fourth step:

- Defining classes in the ontology
- Arranging the classes in an IS-A (subclass–superclass) hierarchy
- Defining attributes of the classes and relationships between the classes.

It is surprising, however, that there is no universal agreement what an ontology actually represents. Schulz et al. (Schulz, Cornet, & Spackman, 2011) in a paper on SNOMED CT's ontological commitment, point out that it is not entirely clear what is represented. They suggest three possible interpretations, which we paraphrase slightly: (1) Concepts represent real-world objects; (2) Concepts represent content elements of Electronic Health Records; (3) Concepts represent patient or clinical situations. Furthermore, if one uses a post-modernist approach to knowledge, it becomes questionable whether any ontology can represent an objective state of the world (Musen, 2014).

In this paper, we propose that under any viewpoint of what ontologies represent, the ontology framework is too limited for pedagogic purposes. We are introducing the idea of an *augmented ontology* that contains additional information related to the use of knowledge in education, which may be seen as a kind of meta-knowledge. We will limit ourselves to the domain of education, leaving open the possibility that augmented ontologies might be helpful in other areas. To concretize the notion of an augmented ontology we note that a "master teacher" knows more than just the course material.

An outstanding teacher who observes that students cannot follow an explanation is able to analyze it for prerequisites that might be unknown to the students. Standard ontologies do not contain prerequisite knowledge. Secondly, an outstanding teacher is also able to prioritize when time is running out. The teacher knows which concepts are essential for the success of the student, and which concepts can be omitted if necessary. Thirdly, an outstanding teacher knows, from years of experience, which concepts are difficult to grasp. Thus, s/he will plan on spending more time concerning difficult concepts.

To reiterate, an ontology augmented for the purpose of education needs to contain prerequisite relationships and difficulty and importance attributes, even if those cannot be objectively established. These three notions are interrelated. For example, a concept with many prerequisites is more likely to be a difficult concept.

In order to transform a domain ontology into an augmented ontology for pedagogic use, heuristics are used, given below. These heuristics are not fail-safe, as heuristics never are, and the final determination has to be made by a domain expert.

H5. A domain concept that occurs many times in a textbook is more important than a concept that occurs few times, especially if the occurrences are spread over a wide range of the book.

H6. A concept that occurs later in a textbook is likely to have more prerequisites than a concept that occurs earlier in the textbook.

H7. A concept that occurs earlier in a book is simpler than a concept that occurs *only* later.

The above heuristics are supported by an index that shows by its page numbers how often and where concepts occur. This information can be extracted algorithmically. We experimented with additional heuristics going beyond the book index, based on concepts appearing in section titles and concepts appearing in the bodies of those sections, however, the success of those heuristics was limited.

4.4 A Tool for Acquiring Knowledge from a Textbook and from an Expert

The SKAT (Security Knowledge Acquisition Tool) was implemented to develop an augmented cyber-security ontology for pedagogic use. Creating this ontology starts by developing a Bootstrap Ontology. Details have been published previously (Wali et al., 2013). The SKAT tool uses information from the index and the book image of a cyber-security textbook to elicit the correct relationships between pairs of a term and a concept. The book by Goodrich and Tamassia was chosen to build the security ontology (Goodrich & Tamassia, 2010). It contains 724 index terms.

We used a glossary of information security terms from the National Institute of Standards and Technology (NIST) Interagency Report (NISTIR) ("Glossary of Key Information Security Terms," 2012) and imported the definitions into the ontology.

Based on the heuristics H1 – H4 from above, SKAT attempts to minimize the effort that the domain expert needs to invest into this process. Furthermore, SKAT suggests the pedagogic knowledge to transform the ontology into an augmented ontology for pedagogic purposes, based on H5 – H7.

Figure 1 shows a screen dump of the implemented SKAT tool. The screen is subdivided into several subwindows. The upper left subwindow presents the projected term list (Definition 7) to the domain expert, ordered by frequency value or in alphabetical order, as desired. The idea is that the user will start with the most commonly co-occurring terms, which, by the above heuristics, are most likely to take part in a semantic relationship of the domain.

Because the IS-A relationship is of paramount importance in ontologies, the domain expert is forced to first connect the chosen term from PT with an IS-A link to a concept in the existing Bootstrap Ontology. The expert can either drag and drop the term into the ontology that appears in the lower right subwindow of the SKAT interface; or s/he can select one of the concepts in the projected concept list PC(T) (Definition 8) for that term in the upper right subwindow.

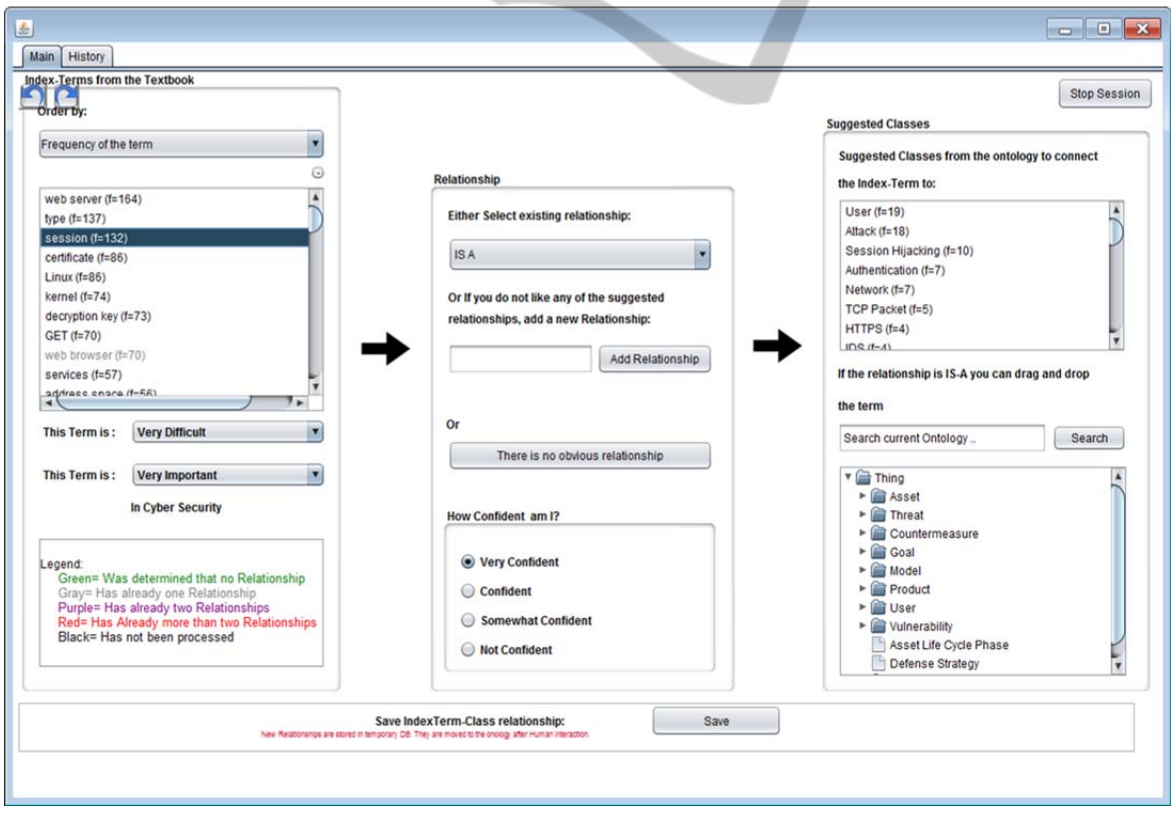

Figure 1: SKAT Tool Client.

Once the term from PT has been connected by an IS-A link to the ontology it is considered promoted to a concept, and the expert may continue with another term, or may assign one or more semantic relationships to the newly created concept. For this purpose s/he may select an already existing relationship from the menu in the upper middle of SKAT or add a new relationship.

The relationship menu is initialized with a list of semantic relationships containing <IS-A, PART-OF, RELATED-TO, KIND-OF>.

SKAT supports color-coding of the number of relationships that connect a term from PT to the ontology. Thus, it is easy to see which terms are still disconnected and which terms are already singly or multiply connected to the ontology.

SKAT supports an undo and a redo button. Changes to the ontology are first saved into a database table and later reviewed by our project team before they are made permanent in the augmented ontology. This allows us to detect and reject any contaminations of the ontology due to accidental misuse of the tool by a domain expert.

A look at the SKAT interface also shows that difficulty and importance are suggested to the domain expert, which he can accept or change. If a domain expert is not confident about a decision s/he may indicate this fact using appropriate buttons.

Domain experts need to log in, in order to have accountability for the decisions made. SKAT records the input of each domain expert together with time stamps, in order to evaluate the time it takes the expert for every decision. SKAT supports an Evaluation and a Production mode.

5 RESULTS

The cyber security ontology is available as a web application at:
http://cis.csi.cuny.edu:8080/Security_OntologyV4/.
SKAT has been made available to two domain experts who have taught classes in network security to evaluate the usability and utility. Their preliminary feedback includes the following:

1. Provide additional functionality. This includes the ability to add any concepts to the ontology, not just those represented as terms in the projected term list.
2. Making corrections to the ontology on the fly should be easier. We note that this functionality was not provided on purpose in order to keep the ontology clean and free of inconsistencies. However, in light of the request we will need to reconsider this design decision.
3. Improve the speed of the implementation. At this time SKAT is the bottleneck, not the decision time of the domain expert. This was surprising for the developers who assumed that the domain experts would need a long time to make decisions.
4. The augmented ontology items interrupt the workflow. It would be better to change SKAT so that the basic ontology building can be separated from the augmentation.
5. The prerequisite augmentation item appears in SKAT in a format that makes it indistinguishable from semantic relationships of the domain. This is confusing.

We are currently working on collecting sessions of the two domain experts and comparing their decisions about the ontology.

6 CONCLUSIONS AND FUTURE WORK

Knowledge is too precious to ignore any sources of it. In this paper we have demonstrated a way of making use of the semi-structured knowledge of a book index and of the book text this index is referring to. However, for many aspects of ontology building, the domain expert is the final arbiter.

While we already have an operational augmented cyber-security ontology, we expect that after several extended sessions of the domain experts the resulting ontology will reflect the textbook knowledge to a high degree and will be helpful as a learning tool in a cyber-security class.

We are currently planning a quantitative and empirical study of the usefulness of the SKAT tool for cyber security domain experts. This involves the evaluation version of SKAT automatically tracking the number of terms classified into the Bootstrap Ontology, the relationships they create and how long each step takes. The security ontology resulting from the hybrid approach, i.e., automated bootstrapping and use of the expert tool, will be used in cyber security education tools e.g., for multimedia course material search, and will be available for other useful applications.

ACKNOWLEDGEMENTS

This work is partially funded by NSF grant DUE1241687. We gratefully acknowledge Pearson

and Addison-Wesley for making the electronic textbooks available. We thank Mickel Mansour for working on and refining the SKAT prototype system.

REFERENCES

An, Y. J., Geller, J., Wu, Y., & Chun, S. A. (2007). *Automatic Generation of Ontology from the Deep Web*. Proceedings Database and Expert Systems Applications. DEXA '07, Regensburg, Germany.

Bajec, M., Eder, J., Souag, A., Salinesi, C., & Comyn-Wattiau, I. (2012). *Ontologies for Security Requirements: A Literature Survey and Classification*. Proceedings Advanced Information Systems Engineering Workshops.

Blanco, C., Lasheras, J., Valencia-Garcia, R., Fernandez-Medina, E., Toval, A., & Piattini, M. (2008). *A Systematic Review and Comparison of Security Ontologies*. Proceedings Third International Conference on Availability, Reliability and Security.

Caracciolo, C. (2006). Designing and Implementing an Ontology for Logic and Linguistics. *Literary & Linguistic Computing, 21*, 29-39.

Chun, S. A., Geller, J., & Wali, A. (2014). *Developing Cyber Security Ontology and Linked Data of Security Knowledge Network*. Proceedings Conference of the Florida Artificial Intelligence Research Society (Flairs-27), Pensacola, FL.

Cimiano, P., Hotho, A., & Staab, S. (2005). Learning concept hierarchies from text corpora using formal concept analysis. *J. Artif. Int. Res., 24*, 305-339.

Cleveland, D. B., & Cleveland, A. D. (2013). *Introduction to indexing and abstracting* (Fourth edition. ed.).

Cornet, R., & de Keizer, N. (2008). Forty years of SNOMED: a literature review. *BMC Med Inform Decis Mak, 8 Suppl 1*, S2. doi: 1472-6947-8-S1-S2 [pii] 10.1186/1472-6947-8-S1-S2

Fellbaum, C. (1998). *WordNet : an electronic lexical database*. Cambridge, Mass: MIT Press.

Fenz, S., & Ekelhart, A. (2009). *Formalizing information security knowledge*. Proceedings 4th International Symposium on Information, Computer, and Communications Security, Sydney, Australia.

Geller, J., Chun, S. A., & An, Y. J. (2008). Toward the Semantic Deep Web. *IEEE Computer*, 95-97.

Geneiatakis, D., & Lambrinoudakis, C. (2007). An ontology description for SIP security flaws. *Comput. Commun., 30*, 1367-1374.

Glossary of Key Information Security Terms. (2012) *NIST Interagency Report* (pp. 222): NIST, US Department of Commerce.

Goodrich, M. t., & Tamassia, R. (2010). *Introduction to Computer Security*: Addison-Wesley.

Hearst, M. A. (1992). *Automatic acquisition of hyponyms from large text corpora*. Proceedings 14th conference on Computational linguistics, Nantes, France.

Herzog, A., Shahmeri, N., & Duma, C. (2007). An Ontology of Information Security. *International Journal of Information Security and Privacy, 1*(4), 1-23.

Hindle, D. (1990). *Noun classification from predicate-argument structures*. Proceedings 28th annual meeting of the Association for Computational Linguistics, Pittsburgh, Pennsylvania.

Humphreys, B. L., & Lindberg, D. A. B. (1993). The UMLS project: making the conceptual connection between users and the information they need. *Bulletin of the Medical Library Association, 81*(2), 170.

Jain, P., Hitzler, P., Sheth, A. P., Verma, K., & Yeh, P. Z. (2010). *Ontology alignment for linked open data*. Proceedings 9th International Semantic Web Conference, Shanghai, China.

Meersman, R., Tari, Z., Kim, A., Luo, J., & Kang, M. (2005). *Security Ontology for Annotating Resources*. Proceedings On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE.

Musen, M. (2014) Personal Communication.

Noy, N. F., & McGuinness, D. L. Ontology Development 101: A Guide to Creating Your First Ontology. From http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html

Pattanasri, N., Jatowt, A., & Tanaka, K. (2007). *Context-aware search inside e-learning materials using textbook ontologies*. Proceedings Joint 9th Asia-Pacific Web and 8th International Conference on web-age information management conference on advances in data and web management, Huang Shan, China.

Schulz, S., Cornet, R., & Spackman, K. (2011). Consolidating SNOMED CT's ontological commitment. *Applied Ontology, 6*(1), 1-11.

Vigna, G., Kruegel, C., Jonsson, E., Undercoffer, J., Joshi, A., & J., P. (2003). Modeling Computer Attacks: An Ontology for Intrusion Detection *Recent Advances in Intrusion Detection* (Vol. 2820, pp. 113-135). Berlin: Springer Verlag.

Wali, A., Chun, S. A., & Geller, J. (2013). *A Bootstrapping Approach for Developing Cyber Security Ontology Using Textbook Index Terms*. Proceedings International Conference on Availability, Reliability and Security (ARES 2013), University of Regensburg, Germany.

Wiebke, P. (2004). A Set-Theoretical Approach for the Induction of Inheritance Hierarchies. *Electron Notes Theor Comput Sci, 53*, 1-13.

Wu, Z., Li, Z., Mitra, P., & Giles, C. L. (2013). *Can Back-of-the-Book Indexes be Automatically Created?* Proceedings CIKM, pp. 1745-1750, San Francisco, CA.